A Note on Approximating the Survivable Network Design Problem in Hypergraphs*

Liang ZHAO†, Nonmember, Hiroshi NAGAMOCHI‡, and Toshihide IBARAKI†, Members

SUMMARY We consider to design approximation algorithms for the survivable network design problem in hypergraphs (SNDPHG) based on algorithms developed for the survivable network design problem in graphs (SNDP) or the element connectivity problem in graphs (ECP). Given an instance of the SNDPHG, by replacing each hyperedge $e = \{v_1, \ldots, v_k\}$ with a new vertex $w_e$ and $k$ edges $\{w_e, v_1\}, \ldots, \{w_e, v_k\}$, we define an SNDP or ECP in the resulting graph. We show that by approximately solving the SNDP or ECP defined in this way, several approximation algorithms for the SNDPHG can be obtained. One of our results is a $d_{\text{max}}$-approximation algorithm for the SNDPHG with $d_{\text{max}}$ ≤ 3, where $d_{\text{max}}$ (resp. $d_{\text{max}}^*$) is the maximum degree of hyperedges (resp. hyperedges with positive cost). Another is a $d_{\text{max}}\mathcal{H}(r_{\text{max}})$-approximation algorithm for the SNDPHG, where $\mathcal{H}(i) = \sum_{j=1}^{i} \frac{1}{j}$ is the harmonic function and $r_{\text{max}}$ is the maximum connectivity requirement.

key words: survivable network design problem, approximation algorithm, connectivity, graph, hypergraph

1. Introduction

A hypergraph is a pair $(V, E)$ of a vertex set $V$ and a hyperedge set $E$, where a hyperedge is a non-empty subset (of any cardinality) of $V$. The degree of a hyperedge is defined as its cardinality. Graphs are hypergraphs such that each hyperedge has degree 2. Hyperedges in graphs are also called edges. Let $s$ and $t$ be two distinct vertices. An $s$-$t$-path is a sequence $v_0 = s, e_1, v_1, \ldots, e_i, v_i, \ldots, e_j, v_j = t$, where $v_i, 0 \leq i \leq j$ (resp. $e_i, 1 \leq i \leq j$), are distinct vertices (resp. hyperedges), such that $v_{i-1}, v_i \in e_i$ holds for all $i = 1, \ldots, j$. Paths are said to be hyperedge-disjoint if no two of them share a common hyperedge.

Given a hypergraph with a nonnegative cost function on hyperedges, and a nonnegativity connectivity requirement $r_{st}$ for each pair of distinct vertices $s$ and $t$, the survivable network design problem in hypergraphs (SNDPHG) asks to find a minimum cost hyperedge subset such that there are at least $r_{st}$ hyperedge-disjoint paths between each pair of vertices $s$ and $t$. The survivable network design problem in graphs (SNDP) is the SNDPHG in graphs. They arise from problems of designing a minimum cost network such that certain vertices are required to remain connected after part of the network fail. An application of the SNDPHG is to compute a minimum cost distribution tree for multicast in communication networks ([7]) involving router costs, where we can model routers by hyperedges. It is known that even the SNDP is NP-hard even for unit costs, where we can model routers by hyperedges. It is known that even the SNDP is NP-hard even for unit costs, where we can model routers by hyperedges. It is known that even the SNDP is NP-hard even for unit costs, where we can model routers by hyperedges. It is known that even the SNDP is NP-hard even for unit costs, where we can model routers by hyperedges. It is known that even the SNDP is NP-hard even for unit costs, where we can model routers by hyperedges.

In the past several years, the SNDP is extensively considered and several approximation algorithms have been developed. Williamson et al. ([1], [2], [9]) have developed a primal-dual method based on algorithms for the SNDPHG with $d_{\text{max}} \leq 3$, where $d_{\text{max}}$ (resp. $d_{\text{max}}^*$) is the maximum degree of hyperedges (resp. hyperedges with positive cost). Another is a $d_{\text{max}}\mathcal{H}(r_{\text{max}})$-approximation algorithm for the SNDPHG, where $\mathcal{H}(i) = \sum_{j=1}^{i} \frac{1}{j}$ is the harmonic function and $r_{\text{max}}$ is the maximum connectivity requirement. For details of this primal-dual method based algorithm, we refer the readers to the well-written survey [4]. Recently, Jain [5] has shown a 2-approximation algorithm for the SNDP based on an iterative rounding process.

Later in [6], Jain et al. considered the element connectivity problem in graphs (ECP). In this problem, the vertex set consists of two types of vertices: terminals and nonterminals. Edges and nonterminals are called the elements. Each pair of terminals has a connectivity requirement, the least number of element-disjoint paths to be realized (where element-disjoint means that no element belongs to two or more paths). The objective is to find a minimum cost edge subset satisfying the requirements. (Notice that only the edges have costs.)

The SNDP is a special case of ECP with an empty nonterminal set. Following the basic algorithmic schema established in [2], [9], they propose an approximation algorithm and claim that it has performance guarantee $2\mathcal{H}(r_{\text{max}})$. We note that, however, the proof in [6] is incomplete, see [10] for details. (Another $2\mathcal{H}(r_{\text{max}})$-approximation algorithm for the ECP with a complete proof can be found in [10].)

It is interesting to know whether these algorithms can be extended to the SNDPHG. Takeshita, Fujito and Watanabe [8] extend the primal-dual approximation algorithm of [3] to the SNDPHG in which $r_{st} \in \{0, 1\}$ for all pair of vertices $s$ and $t$. They show that it is a $d_{\text{max}}$-approximation algorithm for the SNDPHG in graphs. They arise from problems of designing a minimum cost network such that certain vertices are required to remain connected after part of the network fail. An application of the SNDPHG is to compute a minimum cost distribution tree for multicast in communication networks ([7]) involving router costs, where we can model routers by hyperedges. It is known that even the SNDP is NP-hard even for unit costs, where we can model routers by hyperedges. It is known that even the SNDP is NP-hard even for unit costs, where we can model routers by hyperedges. It is known that even the SNDP is NP-hard even for unit costs, where we can model routers by hyperedges.

---

*This research was partially supported by the Scientific Grant-in-Aid from Ministry of Education, Science, Sports and Culture of Japan.

†Dept. of Applied Mathematics and Physics, Graduate School of Informatics, Kyoto University, Kyoto, 606-8501, Japan. Email: {zhao, iibaraki}@amp.i.kyoto-u.ac.jp.

‡Dept. of Information and Computer Sciences, Toyohashi University of Technology, Toyohashi, Aichi, 441-8580, Japan. Email: naga@ics.tut.ac.jp.
approximation algorithm, where $d_{\text{max}}^+$ is the maximum degree of hyperedges. The authors [10] considered the SNDPHG with arbitrary requirements $r_{st}$. Following the basic algorithmic schema established in [2],[9], it is shown that a $d_{\text{max}}^+\mathcal{H}(r_{\text{max}})$-approximation algorithm can be obtained, where $d_{\text{max}}^+ (\leq d_{\text{max}})$ is the maximum degree of hyperedges with positive cost. Moreover, it is shown that the ECP is a special case of SNDPHG in which $d_{\text{max}}$ equals the maximum degree of nonterminals (in ECP) and $d_{\text{max}}^+ = 2$. Thus the algorithm can be viewed as a $2\mathcal{H}(r_{\text{max}})$-approximation algorithm for the ECP. However, it is still open to design an approximation algorithm for the SNDP or ECP such that the performance guarantee is independent of $r_{\text{max}}$.

In this paper, we consider to design approximation algorithms for the SNDPHG on the basis of the algorithms designed for SNDP or ECP. Our approach is very simple. Given an instance of SNDPHG, we replace each hyperedge $e = \{v_1, \ldots, v_k\}$ with a new vertex $w_e$ and $k$ edges $\{w_e, v_1\}, \ldots, \{w_e, v_k\}$. Then an SNDP or ECP is defined in the resulting graph. By approximately solving the SNDP or ECP defined in this way, we can get approximation algorithms for the SNDPHG. We show that given an $\alpha$-approximation algorithm for the SNDP, a $\frac{d_{\text{max}}^+\alpha}{2\mathcal{H}(r_{\text{max}})}$-approximation algorithm for the SNDPHG with $d_{\text{max}} \leq 3$ can be obtained. This result contains the first $r_{\text{max}}$-independent, $d_{\text{max}}^+$-approximation algorithm for the SNDP when $d_{\text{max}} \leq 3$ (using the $\alpha = 2$ result of [5]). We also show that given a $\beta$-approximation algorithm for the ECP, we can get a $\frac{d_{\text{max}}^+\beta}{2\mathcal{H}(r_{\text{max}})}$-approximation algorithm for the SNDPHG. Thus we can get another $d_{\text{max}}^+\mathcal{H}(r_{\text{max}})$-approximation algorithm for the SNDPHG by using the $\beta = 2\mathcal{H}(r_{\text{max}})$ result of [6],[10]. Our approach can also be applied to the problem in which the connectivity requirement is given by a weakly supermodular function, provided several conditions are satisfied (which are satisfied by the SNDPHG). The performance guarantees are then $d_{\text{max}}^+\alpha$ and $d_{\text{max}}^+\beta$ respectively.

2. Algorithms for the SNDPHG

All hypergraphs treated in this paper are undirected. Let $H$ be a hypergraph, and let $V(H)$ and $E(H)$ denote the vertex set and hyperedge set of $H$, respectively. A hyperedge $e$ with endpoints $v_1, \ldots, v_k$ is denoted by $e = \{v_1, \ldots, v_k\}$ and it may be treated as the set $\{v_1, \ldots, v_k\}$ of vertices. The subgraph of $H$ induced by an $S \subseteq V(H)$ is denoted by $H[S] \triangleq (S, E \cap 2^S)$ (where $2^S$ is the power set of $S$, i.e. the set of all subsets of $S$). The neighbors of $S$ in $H$ is denoted by $\Gamma_H(S) \triangleq \{v \in V(H) - S \mid \exists \epsilon \in E(H), v \in \epsilon, \epsilon \cap S \neq \emptyset\}$.

The set of hyperedges in $H$ and $S$ is denoted by $\delta_H(S) \triangleq \{e \in E(H) \mid \emptyset \neq e \cap S \neq \emptyset\}$. Hence $\delta_H(S) = \delta_H(V(H) - S)$. We may use $\Gamma(S)$ and $\delta(S)$ instead of $\Gamma_H(S)$ and $\delta_H(S)$ unless confusion arises.

Instead of treating hypergraphs explicitly, we consider a problem $\mathcal{P}$ in a bipartite graph $G$ which is equivalent to the SNDPHG in a hypergraph $H$. For this, we replace each hyperedge $e = \{v_1, \ldots, v_k\} \in E(H)$ with a new vertex $w_e$ and $k$ edges $\{w_e, v_1\}, \ldots, \{w_e, v_k\}$, setting $w_e$ has the same cost as $e$. To distinguish the newly created vertices from the originals in $V(H)$, we call the original vertices in $V(H)$ as terminals and the new ones as nonterminals, respectively. Notice that there is a one to one correspondence between hyperedges (in hypergraph $H$) and nonterminals (in bipartite graph $G$) with the same degree and cost. Thus in the following, we also use $d_{\text{max}}$ and $d_{\text{max}}^+$ to denote the maximum degree of nonterminals and the maximum degree of nonterminals with positive cost respectively. Formally problem $\mathcal{P}$ is defined as follows.

**Definition 1** (Problem $\mathcal{P}$): Let $G = (T, W, E)$ be a bipartite graph with two disjoint vertex sets $T$ and $W$ and an edge set $E \subseteq \{(v, w) \mid v \in T, w \in W\}$, where vertices in $T$ and $W$ are called terminals and nonterminals respectively. Let $c : W \rightarrow \mathbb{R}^+$ be a nonnegative nonterminal cost function. Given connectivity requirements $r_{st}$ for each pair of distinct terminals $s, t \in T$, find a minimum cost subset $W^* \subseteq W$ such that $G[T \cup W^*]$ has at least $r_{st}$ $W$-disjoint paths between each pair of $s, t \in T$, where $W$-disjoint means that no nonterminal (i.e. vertex in $W$) belongs to two or more paths.

Without loss of generality we assume that $r_{st} = r_{ts}$ for all $s, t \in T$. Notice that there is no connectivity requirement for nonterminals. Given an instance $(G = (T, W, E), c, r)$ of problem $\mathcal{P}$, in order to use the approximation algorithms for the SNDP or ECP (that has costs only on edges), we first define an edge cost function $\bar{c}$ on $E$ according to the cost function $c$ on $W$.

**Definition 2** (cost $\bar{c}$): For each nonterminal $w \in W$ with degree $d_w$, associate each edge $e \in \delta_G((w))$ with cost $\frac{\bar{c}}{d_w}$. Let $\bar{c}$ denote this edge cost function.

2.1 Algorithm for the SNDPHG with $d_{\text{max}} \leq 3$

We show that the SNDPHG with $d_{\text{max}} \leq 3$ can be approximated within factor $d_{\text{max}}^+$. Let us consider the equivalent problem $\mathcal{P}$.

**Algorithm 1**

1. Define an SNDP in $G = (T, W, E)$ with edge cost function $\bar{c}$ and connectivity requirements $r_{st}$ for each pair of distinct terminals $s, t \in T$.
2. Apply an approximation algorithm (for the SNDP) to it. Let $E' \subseteq E$ be the edge subset output.
3. Output $W' = \{w \in W \mid \exists e_1 \neq e_2, s.t. e_1, e_2 \in E' \cap \delta_G((w))\}$.

Clearly Algorithm 1 runs in polynomial time. We next show the correctness and performance guarantee.
**Theorem 1:** If \( d_{\text{max}} \leq 3 \), then Algorithm 1 outputs a feasible solution \( W' \) to Problem \( \mathcal{P} \). Suppose that an \( \alpha \)-approximation algorithm is used in the second step. Then the cost of \( W' \) is at most \( \frac{d_{\text{max}} \alpha}{2} \) times of the optimum.

**Proof.** For any two distinct terminals \( s, t \in T \), there are at least \( r_{st} \) edge-disjoint paths in graph \((T, W, E')\) between \( s \) and \( t \) since \( E' \) is feasible to the SNDP defined in the first step. Notice that for any nonterminal \( w \in W \) that belongs to an \( s, t \)-path, there are exactly two edges in the path that are incident to \( w \) (i.e., in \( \delta_G(\{w\}) \)). Because each nonterminal has degree at most 3 by assumption, we see that these edge-disjoint \( s, t \)-paths are also \( W \)-disjoint. By the definition of \( W' \), this means that there are at least \( r_{st} \) \( W \)-disjoint paths in graph \( G[T \cup W'] \) between each pair of terminals \( s, t \in T \). Thus \( W' \) is feasible to Problem \( \mathcal{P} \).

Let \( W^* \) be an optimal solution to Problem \( \mathcal{P} \). Let \( E^* = \{ e \in E \mid e \cap W^* \neq \emptyset \} \). Notice that \( c(E^*) = c(W^*) \) by the definition of \( c \). Since \( G \) is a bipartite graph, we see that \( W \)-disjoint paths are also edge-disjoint. Hence \( E^* \) is feasible to the SNDP defined in the first step.

Because we use an \( \alpha \)-approximation algorithm in the second step, the cost of \( E' \) is at most \( \alpha \) times of the cost of \( E^* \), i.e., \( c(E') \leq \alpha c(E^*) = \alpha c(W^*) \). By the definition of \( W' \), a nonterminal \( w \) with cost \( c_w \) is included in \( W' \) if and only if \( w \) has at least two incident edges (with cost \( \frac{c_w}{2} \) ) in \( E' \). Thus the cost of \( W' \) is at most \( \frac{d_{\text{max}} \alpha}{2} \) times of \( E' \) (notice that the zero-cost nonterminals has no effect on the cost of \( W' \)). Hence \( c(W') \leq \frac{d_{\text{max}} \alpha}{2} c(W^*) \), which completes the proof. \( \square \)

In particular, we can apply Jain’s 2-approximation algorithm [5] for the SNDP in the second step. Since Problem \( \mathcal{P} \) is equivalent to the SNDPHG in which the degree of a nonterminal equals the degree of the corresponding hyperedge, we have the next corollary.

**Corollary 1:** The SNDPHG with \( d_{\text{max}} \leq 3 \) can be approximated within factor \( d_{\text{max}} \).

Since it has been shown in [10] that the ECP is a special case of the SNDPHG in which \( d_{\text{max}} = 2 \) holds and \( d_{\text{max}} \) corresponds to the maximum degree of nonterminals (in the ECP), we have the next corollary.

**Corollary 2:** The ECP in which nonterminals have degree at most 3 can be approximated within factor 2.

2.2 Algorithm for the SNDPHG with arbitrary \( d_{\text{max}} \)

In this subsection, we show how to approximate the SNDPHG with arbitrary \( d_{\text{max}} \) using an approximation algorithm for the ECP.

**Algorithm 2**

1. Define an ECP in \( G = (T, W, E) \) with terminal set \( T \), nonterminal set \( W \), edge cost function \( c \) and connectivity requirements \( r_{st} \) for each pair of distinct terminals \( s, t \in T \).
2. Apply an approximation algorithm (for the ECP) to it. Let \( E'' \subseteq E \) be the edge subset output.
3. Output \( W'' = \{ w \in W \mid \exists e_1 \neq e_2, \text{ s.t. } e_1, e_2 \in E'' \cap \delta_G(\{w\}) \} \).

It is clear that Algorithm 2 runs in polynomial time.

**Theorem 2:** Algorithm 2 outputs a feasible solution \( W'' \) to Problem \( \mathcal{P} \). Suppose that a \( \beta \)-approximation algorithm is used in the second step. Then the cost of \( W'' \) is at most \( \frac{d_{\text{max}} \beta}{2} \) times of the optimum.

**Proof.** Since \( E'' \) is feasible to the ECP defined in the first step, we see that for any two distinct terminals \( s, t \in T \), there are at least \( r_{st} \) element-disjoint paths in graph \((T, W, E'')\) between \( s \) and \( t \), where element stands for an edge or a nonterminal. Clearly element-disjoint paths are also \( W \)-disjoint since \( W \) is a subset of the element set \( W \cup E \). Notice that for any nonterminal \( w \in W \) that belongs to an \( s, t \)-path, there are exactly two edges in the path that are incident to \( w \) (i.e., in \( \delta_G(\{w\}) \)). Thus \( W'' \) is feasible to Problem \( \mathcal{P} \).

Let \( W^* \) be an optimal solution to Problem \( \mathcal{P} \). Let \( E^* = \{ e \in E \mid e \cap W^* \neq \emptyset \} \). Since \( W \)-disjoint paths are also edge-disjoint, thus element-disjoint, we see that \( E^* \) is feasible to the ECP defined in the first step. Similarly to the proof of Theorem 1, it is easy to show the performance guarantee \( \frac{d_{\text{max}} \beta}{2} \).

In particular, we can apply the \( 2\mathcal{H}(r_{\text{max}}) \)-approximation algorithm for the ECP ([6], [10]) in the second step, which yields the next corollary.

**Corollary 3 ([10]):** The SNDPHG can be approximated within factor \( d_{\text{max}} \mathcal{H}(r_{\text{max}}) \).

We note that under some assumptions (which are satisfied by the SNDPHG), the algorithm in [10] is a \( d_{\text{max}} \mathcal{H}(r_{\text{max}}) \)-approximation algorithm for more general problems in which the connectivity requirements are given by a weakly supermodular function.

3. On the Generalizations

In fact, all the papers [1], [2], [5], [9], [10] have considered problems more general than the SNDP, ECP or SNDPHG, where the connectivity requirements are generalized by a requirement function. In this section, we consider a generalization of our approach in Sect. 2.

In the generalization of SNDP, the requirements \( r_{st} \) for each pair of vertices \( s \) and \( t \) are generalized by a connectivity requirement function \( r : 2^V \rightarrow \mathbb{Z}^+ \), where \( V \) is the vertex set and \( \mathbb{Z}^+ \) is the set of nonnegative integers. The objective is to find a minimum cost edge subset \( E^* \).
such that $|\delta(S) \cap E^*| \geq r(S)$ for all $S \subseteq V$. It is known that the SNDP can be formulated in this way with $r(S) = \max \{r_{st}, s \in S, t \notin S\}$ for all $S \subseteq V$ ([4]). The algorithms in [1], [2], [5], [9] can be applied to this generalization of SNDP, provided that certain conditions are satisfied (which are satisfied by the SNDP). One of such conditions is that $r$ is weakly supermodular, i.e., $r(V) = 0$ and for any two sets $A, B \subseteq V$, $r(A - B) \leq r(A) - r(B)$ or $r(A \cup B) = r(A) + r(B)$ ($r(A \cap B)$). Other conditions are complicated and vary with algorithms, thus are left to later when we use them.

We want to generalize our approach. For this, let us consider the next generalization of Problem $P$.

**Definition 3 (Problem $\overline{P}$):** Let $G = (T, W, E)$ be a bipartite graph with terminal set $T$, nonterminal set $W$ and edge set $E$. Let $c : W \rightarrow \mathbb{R}^+$ be a nonnegative nonterminal cost function. Let $r : 2^T \rightarrow \mathbb{Z}^+$ be a weakly supermodular connectivity requirement function. Find a minimum cost subset $W^* \subseteq W$ such that

$$|\Delta(S) \cap W^*| \geq r(S) \quad \forall S \subseteq T,$$

where $\Delta(S) = \Gamma(S) \cap \Gamma(T - S)$ (with respect to $G$).

It has been shown that Problem $P$, hence the SNDPHG, can be formulated in this way with $r$ defined by $r(S) = \max \{r_{st}, s \in S, t \in T - S\}$ for all $S \subseteq T$ ([10]). In the following, we show two approximation algorithms to the Problem $\overline{P}$.

3.1 Generalization of Algorithm 1

Consider a function $\tilde{r} : 2^{T \cup W} \rightarrow \mathbb{Z}^+$ defined by

$$\tilde{r}(S) = r(S \cap T) \quad \forall S \subseteq T \cup W.$$  

(2)

It is clear that function $\tilde{r}$ is also weakly supermodular by the weak supermodularity of $r$. Given an instance $(G = (T, W, E), c, r)$ of Problem $\overline{P}$ with $d_{\text{max}} \leq 3$, we consider the next algorithm.

**Algorithm 3**

1. For a bipartite graph $G = (T, W, E)$ with edge cost function $\tilde{c}$ (see Definition 2) and connectivity requirement function $\tilde{r}$ (defined by (2)), define a problem of finding a minimum cost edge subset $E^* \subseteq E$ such that

$$|\delta(S) \cap E^*| \geq \tilde{r}(S) \quad \forall S \subseteq T \cup W.$$  

(3)

2. Apply an algorithm to the problem defined in the first step. Let $E' \subseteq E$ be the edge subset output.

3. Output $W' = \{w \in W \mid \exists e \in E' \cap \delta_G(\{w\})\}$.

Note that we can apply one of the algorithms in [1], [2], [5], [9] in the second step, provided that the conditions needed by such algorithm are satisfied. Notice that since $\overline{P}$ has no longer a notion of paths as used before, the third step is different from Algorithm 1 and 2.

Clearly, Algorithm 3 runs in polynomial time if so does the algorithm used in the second step. We show the performance guarantee in the next theorem.

**Theorem 3:** If $d_{\text{max}} \leq 3$, then Algorithm 3 outputs a feasible solution $W'$ to Problem $\overline{P}$. Suppose that an $\alpha$-approximation algorithm is used in the second step. Then the cost of $W'$ is at most $d_{\text{max}}^2 \alpha$ times of the optimum.

**Proof.** We first show that $W'$ is feasible to Problem $\overline{P}$, i.e., $|\Delta(S) \cap W'| \geq r(S)$ for all $S \subseteq T$. For this, let $S = \{w \in \Delta(S) \mid \exists t_1, t_2 \in S, s.t. t_1 \neq t_2 \in \{t_1, w\}, \{t_2, w\}, \{t_1, w\}, \{t_2, w\} \in E\}$. It is not difficult to see that $|\delta(S) \cap E'| = |\Delta(S) \cap W'|$ (notice that any nonterminal has degree at most 3). Therefore we have $|\Delta(S) \cap W'| = |\delta(S) \cap E'| \geq \tilde{r}(S) = r(S)$.

Similarly to the proof of Theorem 1, we can derive the performance $d_{\text{max}}^2 \alpha$. \hfill $\square$

**Corollary 4:** If $d_{\text{max}} \leq 3$ and there is a polynomial time separation oracle for the LP relaxation of the problem defined in the first step, then Algorithm 3 is a $2d_{\text{max}}^2$-approximation algorithm by using Jain’s 2-approximation algorithm ([5]) in the second step.

**Proof.** Jain’s algorithm needs two conditions. One is that there is a polynomial time separation oracle for the LP relaxation, which is satisfied by assumption. Another is that $\tilde{r}$ must be a weakly supermodular function, which we have already seen. \hfill $\square$

3.2 Generalization of Algorithm 2

We next show that Algorithm 2 can be extended.

In [10], the authors have shown that Problem $\overline{P}$ with $d_{\text{max}}^2 = 2$ is a generalization of ECP, and can be approximated within factor $2\Theta(r_{\text{max}})$ provided that the so-called minimal violated sets can be found in polynomial time. We want to use this result to generalize Algorithm 2 to Problem $\overline{P}$. Since the costs are only on nonterminals in [10], $\tilde{c}$ (Definition 2) cannot be used. We define a new problem in the following way.

Suppose an instance $(G = (T, W, E), c, r)$ of Problem $\overline{P}$ is given. For each nonterminal $w \in W$ with cost $c_w > 0$ and degree $d_w \geq 3$, replace each edge $e = \{t, w\} \in E$ with three new edges $e_1 = \{t, a_{tw}\}, e_2 = \{a_{tw}, b_{tw}\}$ and $e_3 = \{b_{tw}, w\}$, introducing a new nonterminal $a_{tw}$ and a new terminal $b_{tw}$. Let the cost of new nonterminals $a_{tw}$ be $\frac{c_e}{c_w}$, and the cost of nonterminals $w$ be zero. Let $\hat{c}$ denote this nonterminal cost function. Let $\tilde{T}, \tilde{W}$ and $\tilde{E}$ denote the new terminal set $\tilde{T} \cup \{b_{tw}\}$, the new nonterminal set $\tilde{W} \cup \{a_{tw}\}$ and the new edge set respectively (where $\{b_{tw}\}$ and $\{a_{tw}\}$ are the sets of all $b_{tw}$ and $a_{tw}$ respectively). Define a connectivity requirement function $\hat{r}$ by $\hat{r}(S) = \tilde{r}(S \cap \tilde{T})$ for all $S \subseteq \tilde{T}$. Clearly $\hat{r}$ is weakly supermodular.

**Algorithm 4**
1. Define a problem (in the formulation of Problem $\mathcal{P}$) in $\tilde{G} = (T, W, \tilde{E})$ with nonterminal cost function $\tilde{c}$ and connectivity requirement function $\tilde{r}$.
2. Apply an algorithm on the problem defined in the first step. Let $W'' \subseteq W$ be the edge subset output.
3. Output $W'' = \{ w \in W \mid \exists t \in T \text{ s.t. } a_{tw} \in W' \}$.

Next theorem shows the correctness and performance guarantee of Algorithm 4, for which the proof is similar to before and is omitted.

**Theorem 4:** Algorithm 4 outputs a feasible solution $W''$ to Problem $\mathcal{P}$. Suppose that an $\beta$-approximation algorithm is used in the second step. Then the cost of $W''$ is at most $d_{\max}^{+} \beta$ times of the optimum.

Notice that in the problem defined in the first step, the maximum degree of nonterminals with positive cost is 2. Thus algorithm [10] is applicable in the second step.

**Corollary 5:** If the minimal violated sets can be found in polynomial time for the problem defined in the first step, then Algorithm 4 is a $2d_{\max}^{+} \mathcal{H}(r_{\max})$-approximation algorithm by using the algorithm of [10] in the second step.

**References**


