Degree Bounded Network Design with Metric Costs

Yuk Hei Chan Wai Shing Fung Lap Chi Lau Chun Kong Yung

Department of Computer Science and Engineering
The Chinese University of Hong Kong

Abstract

Given a complete undirected graph, a cost function on edges and a degree bound B, the degree bounded network design problem is to find a minimum cost simple subgraph with maximum degree B satisfying given connectivity requirements. Even for simple connectivity requirement such as finding a spanning tree, computing a feasible solution for the degree bounded network design problem is already NP-hard, and thus there is no polynomial factor approximation algorithm for this problem. In this paper, we show that when the cost function satisfies triangle inequalities, there are constant factor approximation algorithms for various degree bounded network design problems.

- Global edge-connectivity: *There is a $(2 + \frac{1}{k})$-approximation algorithm for the minimum bounded degree k-edge-connected subgraph problem.*
- Local edge-connectivity: *There is a 6-approximation algorithm for the minimum bounded degree Steiner network problem.*
- Global vertex-connectivity: *There is a $(2 + \frac{k-1}{n} + \frac{1}{k})$-approximation algorithm for the minimum bounded degree k-vertex-connected subgraph problem.*
- Spanning tree: *There is an $(1 + \frac{1}{d-1})$-approximation algorithm for the minimum bounded degree spanning tree problem.*

These approximation algorithms return solutions with smallest possible maximum degree, and the cost guarantee is obtained by comparing to the optimal cost when there are no degree constraints. This demonstrates that degree constraints can be incorporated into network design problems with metric costs.

Our algorithms can be seen as a generalization of Christofides' algorithm for metric TSP. The main technical tool is a simplicity-preserving edge splitting-off operation, which is used to "short-cut" vertices with high degree while maintaining connectivity requirements and preserving simplicity of the solutions.

1 Introduction

Consider the problem of finding a minimum cost k-edge-connected subgraph with maximum degree at most B in a weighted undirected graph. This is a generalization of the travelling salesman problem when $k = B = 2$, and the minimum bounded degree spanning tree problem when $k = 1$. In general this problem does not admit any polynomial time approximation algorithm, since the feasibility problem is already NP-hard. Recent research has thus focused on obtaining bicriteria approximation algorithms for degree bounded network design problems [18, 28, 36, 29].

In some network design problems the cost function satisfies triangle inequalities, and stronger algorithmic results are known [24, 9, 11]. For the traveling salesman problem, although there is no polynomial factor approximation algorithms in general, it is well-known that there is a 1.5-approximation algorithm assuming triangle inequalities [10]. This motivates us to study more general degree bounded network design problems with metric costs.

Formally, we are given an undirected graph $G = (V, E)$, a connectivity requirement function $r : V \times V \to \mathbb{Z}$ on pair of vertices, a cost function $c : E \to \mathbb{Q}$ on edges satisfying triangle inequalities ($c(uv) + c(vw) \geq c(uw)$ for all u, v, w), and a degree upper bound B on each vertex v. The goal is to find a minimum cost subgraph $H \subseteq G$ that has at least $r(u, v)$ edge-disjoint (or vertex-disjoint) paths between u and v, and the degree of each vertex in H is at most B.

1.1 Results

We show that there are constant factor approximation algorithms for various degree bounded network design problems with metric costs. In addition, these algorithms return solutions with smallest possible maximum degree (e.g. k-connected subgraphs with maximum degree k) and the cost is within a constant time the optimal cost when there are no degree constraints. This demonstrates that degree constraints can be incorporated into network design problems with metric costs.

Global Edge-Connectivity: We first consider the problem of finding a minimum cost k-edge-connected simple subgraph with metric costs. The main procedure is to transform any k-edge-connected simple subgraph into a k-edge-connected simple subgraph with maximum degree k, with only a small increase in the cost.

Theorem 1.1. *Given a complete graph $G = (V, E)$ with metric costs and any simple k-edge-connected subgraph H of G, there is a polynomial time algorithm to construct:*

1. *A simple k-edge-connected subgraph H' with maximum degree at most $k + 1$ and $cost(H') \leq cost(H)$.*

2. *A simple k-edge-connected subgraph H'' with maximum degree k and $cost(H'') \leq cost(H) + EC_k(G)/k$, where $EC_k(G)$ is the cost of a minimum k-edge-connected subgraph of G.* [1]

We remark that if parallel edges are allowed in the solutions, then a similar statement as Theorem 1.1(1) is proved by Bienstock, Brickell, Monma [6]. However, for degree bounded network design problems, there are capacity constraints on edges and so their result can not be directly applied.[2] Theorem 1.1 implies the first constant factor approximation algorithm for the minimum bounded degree k-edge-connected subgraph problem.

Theorem 1.2. *Given a complete graph with metric costs, there is a polynomial time $(2 + 1/k)$-approximation algorithm for the minimum bounded degree k-edge-connected subgraph problem.*

Local Edge-Connectivity: Theorem 1.1 can be extended to general edge-connectivity requirements. In the following let $r_{\max} := \max_{u,v} r(u, v)$ be the maximum edge-connectivity requirement, and call a subgraph H satisfying all connectivity requirements a Steiner network.

Theorem 1.3. *Given a complete graph $G = (V, E)$ with metric costs and any simple Steiner network H of G, there is a polynomial time algorithm to construct:*

1. *A simple Steiner network H' with maximum degree at most $r_{\max} + 1$ and $cost(H') \leq 2 \cdot cost(H)$.*

2. *A simple Steiner network H'' with maximum degree at most $r_{\max}$ and $cost(H'') \leq 3 \cdot cost(H)$, when $r_{\max}$ is even.* [3]

In the following we say an algorithm is an $(\alpha, +\beta)$-bicriteria approximation algorithm if it returns a solution with cost at most $\alpha \cdot \text{OPT}$ and the degree of each vertex is at most $B + \beta$. Theorem 1.3 implies the first constant factor approximation algorithm for the minimum bounded degree Steiner network problem with metric costs.

Theorem 1.4. *Given a complete graph with metric costs, there is a polynomial time $(4, +1)$-approximation algorithm for the minimum bounded degree Steiner network problem. For $r_{\max}$ even, there is a polynomial time 6-approximation algorithm for the minimum bounded degree Steiner network problem.* [3]

Global Vertex-Connectivity: Similar result can be obtained for vertex-connectivity, with the additional requirement that $|V| \geq 2k$. Note that the first part of the following theorem is proved by Bienstock, Brickell and Monma [6].

Theorem 1.5. *Given a complete graph $G = (V, E)$ with metric costs and a k-vertex-connected subgraph H of G with $|V| \geq 2k$, there is a polynomial time algorithm to construct:*

1. *[6] A k-vertex-connected subgraph H' with maximum degree at most $k + 1$ and $cost(H') \leq cost(H)$.*

2. *A k-vertex-connected subgraph H'' with maximum degree k and $cost(H'') \leq cost(H) + VC_k(G)/k$, where $VC_k(G)$ is the cost of a minimum k-vertex-connected subgraph of G.* [1]

This implies the first constant factor approximation algorithm for the minimum bounded degree k-vertex-connected subgraph problem with metric costs. Note that without the metric cost assumption, there is no known constant factor approximation algorithms for the minimum cost k-vertex-connected subgraph problem, and the degree bounded k-vertex-connected subgraph problem.

Theorem 1.6. *For $|V| \geq 2k$, there is a $(2 + \frac{k-1}{n} + \frac{1}{k})$-approximation algorithm for the minimum bounded degree k-vertex-connected subgraph problem.*

Spanning Tree: There is a simple 2-approximation algorithm for the minimum bounded degree spanning tree problem with metric costs. Improvements are known for special metric costs such as Euclidean distances [34, 22, 21, 7], but not known for general metric costs. The following result improves upon the simple 2-approximation algorithm for all $d \geq 3$. [4]

[1] When both k and $|V|$ are odd numbers, then it is impossible to have a k-regular-subgraph. In that case our algorithm can choose any vertex v in the graph, and returns a solution with v having degree $k + 1$ while all other vertices having degree k, which is best possible.

[2] Incidentally, if parallel edges are allowed, then there is a simple constant factor approximation algorithm by taking $k/2$ copies of an approximate solution of metric TSP.

[3] When $r_{\max}$ is odd, each connected component may have one vertex with degree $r_{\max} + 1$.

[4] For $d = 2$ Christofides' algorithm is a 3/2-approximation algorithm for the minimum bounded degree spanning tree problem with metric costs.

Theorem 1.7. *Given a complete graph with metric costs, there is a polynomial time algorithm to find a spanning tree with maximum degree* d *whose cost is at most* $1 + \frac{1}{d-1}$ *times the cost of an optimal solution with maximum degree* d*.*

1.2 Techniques

Our algorithms can be seen as a generalization of Christofides' algorithm for metric TSP. Christofides' algorithm first constructs a minimum spanning tree, then adds a minimum perfect matching between odd degree vertices, and finally short-cuts the high degree vertices without increasing the cost. The approach taken in this paper is similar. We illustrate it in the global edge-connectivity setting. First we construct a k-edge-connected subgraph H (without degree constraints) by using an existing 2-approximation algorithm for the minimum cost k-edge-connected subgraph problem [23]. Then we apply a short-cutting procedure to transform H into a k-edge-connected subgraph H' of maximum degree $k+1$ without increasing the cost. Finally we add a minimum cost perfect matching to vertices with degree $k+1$ in H', and then apply the short-cutting procedure once again to transform it to a k-edge-connected subgraph H'' of maximum degree k.

To short-cut the high degree vertices, we use the edge splitting-off operation, which involves replacing two edges xu and xv sharing the same vertex x by the edge uv. With the metric cost assumption, the new edge uv is no more expensive than the cost of xu and xv, and so this operation can be used to decrease the degree of x by 2 without increasing the cost. However, in general the connectivity requirements may be violated after an edge splitting-off operation is performed. The first edge splitting-off result is proved by Lovász [30], where he gave sufficient conditions for the existence of an edge splitting-off operation that maintains global edge-connectivity. This result has been extended in different directions [31, 32, 6, 2, 20] and has found a number of applications in graph connectivity problems, including connectivity augmentation [13, 3], graph orientation [30, 14], Steiner tree packing [26, 27], etc.

We are concerned with the simplicity of the solutions, and so require a *simplicity-preserving* edge splitting-off operation that maintains edge-connectivity and does not introduce new parallel edges. Simplicity preserving edge splitting-off was studied by Bang-Jensen and Jordán [3], where they show that if the degree of a vertex v is at least $\Omega(k^4)$, then there is a complete edge splitting-off on v (i.e. split-off all the edges incident to v) that preserves simplicity and maintains k-edge-connectivity of the remaining graph. Their result is applied to the simplicity preserving connectivity augmentation problem, and gives an approximation algorithm which adds at most $O(k^2)$ more edges than the optimal solution.

For degree bounded network design problems, there is no need for a complete edge splitting-off, and we prove a sharper degree bound for the existence of an appropriate simplicity-preserving edge splitting-off operation. Our main technical result is Theorem 2.2, which roughly says: if the degree of a vertex v is at least $r_{\max} + 2$, then there is a simplicity-preserving edge splitting-off operation that maintains local edge-connectivity requirements for all pairs. As a by-product, this also gives a new proof of Mader's theorem (see Theorem 2.1) on edge splitting-off maintaining local edge-connectivity.

The strategy for local edge-connectivity and global vertex-connectivity is similar. We remark that the procedure of reducing maximum degree by edge splitting-off operation was first used by Bienstock, Brickell and Monma [6], where they also proved the first edge splitting-off result maintaining global vertex-connectivity (see Theorem 3.1), and a similar result as Theorem 1.1(1) when parallel edges are allowed. For spanning trees, our result is obtained by combining a recent bicriteria result by Singh and Lau [36] and a minimum cost flow technique by Fekete et.al. [12].

1.3 Related Work

Network design problems with metric costs are well-studied problems in the literature [24, 9, 25, 11]. Here we focus on related work on degree bounded network design problems. For general cost function, a polyhedral approach is applied successfully to obtain bicriteria approximation algorithms with only an additive violation on the degree: there is a $(1, +1)$-approximation algorithm for the minimum bounded degree spanning tree problem [36], a $(2, +O(k))$-approximation algorithm for the minimum bounded degree k-edge-connected subgraph problem [29], and a $(2, +O(r_{\max}))$-approximation algorithm for the minimum bounded degree Steiner network problem [29], while the maximum degree of the solutions is at most $2B+3$ [28].

For bounded degree network design problem with metric costs, there are approximation algorithms to construct a k-edge-connected subgraph with maximum degree k [16, 17], if parallel edges are allowed. For local edge-connectivity, there is some known result [15], but no constant factor approximation algorithm is known even if parallel edges are allowed. For vertex-connectivity, it was first studied in [33] for 2-vertex-connectivity, in which they showed that there exists an optimal solution with maximum degree 3. This result has been generalized to k-vertex-connectivity in [6], which also implies a bicriteria $(2 + \frac{k-1}{n}, +1)$-approximation algorithm for the minimum bounded degree k-vertex-connected subgraph problem. For bounded degree spanning trees, there is a simple 2-approximation algorithm, and improvement over this 2-approximation algorithm was known for Euclidean space [34, 12, 22, 1, 7, 21].

2 Simplicity-Preserving Edge Splitting-Off

The splitting-off operation involves replacing two edges xu and xv sharing the same vertex x by the edge uv. The main content of edge splitting-off results is to maintain the edge-connectivity of the graph. Lovász [30] obtained the first splitting-off result concerning global edge-connectivity of the resulting graph, and Mader [31] extended it to local edge-connectivity. The *local edge-connectivity* between two vertices u and v is defined as the maximum number of edge-disjoint paths between u and v.

Theorem 2.1. **(Mader's theorem)** *If $d(x) \neq 3$ and there is no cut edge incident to x, then there is an edge splitting-off operation on x that maintains the local edge-connectivity for every pair of vertices $u, v \in V - x$.*

We consider *simplicity-preserving* edge splitting-off that does not introduce new parallel edges, that is, we do not allow splitting off xu, xv if the edge uv already exists. This was first studied by Bang-Jensen and Jordán [3], where they proved that if $d(x) = \Omega(k^4)$, then there is a complete splitting-off on x that maintains k-edge-connectivity in the remaining graph. For degree bounded network design problems, there is no need for a complete edge splitting-off. Our main technical result provides sufficient conditions that guarantee the existence of a simplicity-preserving edge splitting-off operation that maintains local edge-connectivity requirements.

Theorem 2.2. *Suppose $N(x)$ is not a clique and $|N(x)| \geq r_{\max} + 2$. If $d(x) \neq 3$ and there is no cut edge incident to x, then there is a simplicity-preserving edge splitting-off operation on x that maintains the local edge-connectivity for every pair of vertices $u, v \in V$.*

2.1 Preliminaries

Let $G = (V, E)$ be a graph. For $X, Y \subseteq V$, denote by $\delta(X, Y)$ the set of edges with one endpoint in $X - Y$ and the other endpoint in $Y - X$ and $d(X, Y) := |\delta(X, Y)|$, and also define $\bar{d}(X, Y) := d(X \cap Y, V - (X \cup Y))$. For $X \subseteq V$, define $\delta(X) := \delta(X, V - X)$ and the *degree* of X as $d(X) := |\delta(X)|$. Denote the degree of a vertex as $d(v) := d(\{v\})$. Also denote the set of neighbours of v by $N(v)$, and call a vertex in $N(v)$ a *v-neighbour*.

Let $r(u, v)$ be the edge-connectivity requirement (number of edge-disjoint paths) between u and v. The requirement $r(X)$ of a set $X \subseteq V$ is the maximum edge-connectivity requirement between u and v with $u \in X$ and $v \in V - X$. By Menger's theorem, to satisfy the connectivity requirements, it suffices to guarantee that $d(X) \geq r(X)$ for all $X \subset V$. The *surplus* $s(X)$ of a set $X \subseteq V$ is defined as $d(X) - r(X)$. A graph satisfies the edge-connectivity requirements if $s(X) \geq 0$ for all $\emptyset \neq X \subset V$. For $X \subseteq V - x$, X is called *tight* if $s(X) = 0$ and *dangerous* if $s(X) \leq 1$, where x is the vertex to be split-off. The following proposition will be used throughout our proof.

Proposition 2.3. *[13] For arbitrary $X, Y \subseteq V$ at least one of the following inequalities holds:*

$$s(X) + s(Y) \geq s(X \cap Y) + s(X \cup Y) + 2d(X, Y) \tag{2.3a}$$

$$s(X) + s(Y) \geq s(X - Y) + s(Y - X) + 2\bar{d}(X, Y) \tag{2.3b}$$

Two edges xu, xv form an *admissible pair* if the graph after splitting-off xu, xv does not violate $s(X) \geq 0$ for all $X \subset V$. An admissible pair is *legal* if no new parallel edge is formed after the pair is split-off. For convenience, when we consider a pair of edges, they are assumed to be incident to x unless otherwise specified. The following proposition characterizes when a pair is admissible.

Proposition 2.4. *[13] A pair xu, xv is not admissible if and only if u, v are both contained in some dangerous set.*

2.2 Proof of Theorem 2.2

Suppose, by way of contradiction, that all the conditions in Theorem 2.2 are satisfied, but there is no legal pair on x. We will prove in Lemma 2.6 that a certain 3-dangerous-set structure exists, see Figure 1(a). Then we will prove in Lemma 2.7 that such a 3-dangerous-set structure would imply that either $d(x) = 3$ or there is a cut edge incident to x, violating the conditions in Theorem 2.2. We remark that Lemma 2.7 can also be used to give a new proof of Mader's theorem.

First we need the following claim to establish the 3-dangerous-set structure.

Claim 2.5. *Suppose $|N(x)| \geq r_{\max} + 2$. Then for any dangerous set D, there exists a vertex $w \in N(x) \cap (V - D)$ with $d(w, D) = 0$.*

Proof. If D contains all x-neighbours, then $d(D) \geq |N(x)| \geq r_{\max} + 2$ and contradicts the assumption that D is dangerous. Therefore $N(x) \cap (V - D) \neq \emptyset$. Each vertex in $N(x) \cap D$ contributes at least one to $d(D)$. Suppose, by way of contradiction, that $d(v, D) \geq 1$ for each $v \in (V - D) \cap N(x)$. Then $d(D) \geq |N(x) \cap D| + |N(x) \cap (V - D)| = |N(x)| \geq r_{\max} + 2$, which contradicts the assumption that D is dangerous. Therefore there exists a vertex $w \in N(x) \cap (V - D)$ with $d(w, D) = 0$. □

The following lemma shows that a certain 3-dangerous-set structure as shown in Figure 1(a) must exist, which is a crucial step in the proof.

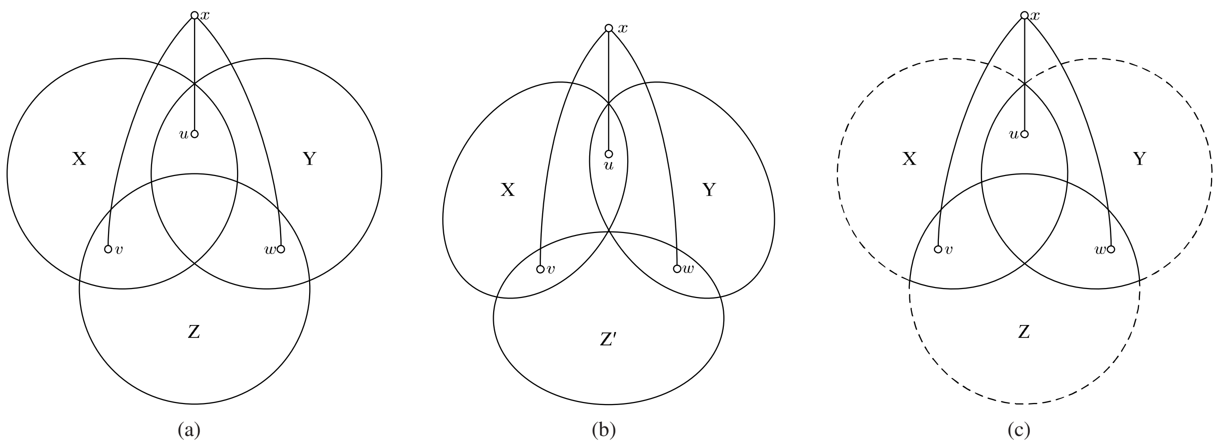

Figure 1. The 3-dangerous-set structures.

Lemma 2.6. *Suppose $N(x)$ is not a clique and $|N(x)| \geq r_{\max} + 2$. If there is no legal pair on x, then there exist maximal dangerous sets X, Y, Z and $u, v, w \in N(x)$ such that $u \in X \cap Y$, $v \in X \cap Z$, $w \in Y \cap Z$ and $u, v, w \notin X \cap Y \cap Z$.*

Proof. Since $N(x)$ is not a clique, there exist $u', v' \in N(x)$ with $u'v' \notin E$. Since there is no legal pair on x, xu', xv' must be non-admissible. By Proposition 2.4, there exists a dangerous set that contains both u' and v'. Let X be a maximal dangerous set containing u', v' such that $X \cap N(x)$ is not a proper subset of $D \cap N(x)$ for any dangerous set D.

By Claim 2.5, there exist $w' \in N(x) \cap (V - X)$ such that $u'w', v'w' \notin E$. As there is no legal pair on x, both (xu', xw') and (xv', xw') must be non-admissible. By Proposition 2.4, there exist a dangerous set containing (u', w') and a dangerous set containing (v', w'). If there exist maximal dangerous sets Y and Z that $u', w' \in Y$, $v' \notin Y$ and $v', w' \in Z$, $u' \notin Z$, then we get the desired 3-dangerous-set structure.

Otherwise, there must exist a maximal dangerous set Y that $u', v', w' \in Y$. Since $\bar{d}(X, Y) \geq d(u' + v', x) \geq 2$, we have $s(X) + s(Y) \leq 1 + 1 < 2\bar{d}(X, Y)$. So inequality (2.3b) cannot hold for (X, Y), and thus inequality (2.3a) must hold for (X, Y). As X is maximally dangerous and $w' \in Y - X$, $X \cup Y$ cannot be dangerous and thus $s(X \cup Y) \geq 2$. Therefore, by inequality (2.3a),

$$\begin{aligned} 1 + 1 &\geq s(X) + s(Y) \\ &\geq s(X \cap Y) + s(X \cup Y) + 2d(X, Y) \geq 2. \end{aligned}$$

This implies that $s(X \cap Y) = d(X, Y) = 0$ and $s(X \cup Y) = 2$. By the definition of X, $X \cap N(x)$ is not a proper subset of $Y \cap N(x)$. Since $w' \in N(x) \cap Y$, there must exist $t' \in N(x) \cap X$ and $t' \notin Y$. Since $d(X, Y) = 0$, we have $t'w' \notin E$. For $\{xt', xw'\}$ to be illegal, there exists a maximal dangerous set Z containing both w' and t'. We will show that both u' and v' are not in Z. By using this, we can define $u = u', w = w', v = t'$ and get the desired 3-dangerous-set structure.

We now complete the proof by showing that both u' and v' are not in Z. Suppose, by way of contradiction, that $u' \in Z$. Then since $\bar{d}(Y, Z) \geq d(u' + w', x) \geq 2$ and $s(Y) + s(Z) \leq 1 + 1 = 2$, inequality (2.3b) does not hold for (Y, Z) and thus inequality (2.3a) must hold for (Y, Z). As Y is a maximal dangerous set and $t' \in Z$, $Y \cup Z$ cannot be a dangerous set and $s(Y \cup Z) \geq 2$. By inequality (2.3a) for (Y, Z), this implies that $s(Y \cup Z) = 2$ and $d(Y, Z) = s(Y \cap Z) = 0$. Consider $Y \cap Z$ and X. Note that $\bar{d}(Y \cap Z, X) \geq d(u', x) \geq 1$ and $s(Y \cap Z) + s(X) \leq 0 + 1 = 1$. So inequality (2.3b) does not hold for $(Y \cap Z, X)$, and thus inequality (2.3a) must hold for $(Y \cap Z, X)$. Therefore we have $s((Y \cap Z) \cup X) \leq s(Y \cap Z) + s(X) = 1$, which implies that $(Y \cap Z) \cup X$ is a dangerous set. Since $w \in (Y \cap Z) - X$, this contradicts the maximality of X and completes the proof. □

The following lemma shows that the 3-dangerous-set structure in Lemma 2.6 (Figure 1(a)) would contradict with the conditions of Theorem 2.2; similar structures also appear in [4, 5]. This will complete the proof of Theorem 2.2.

Lemma 2.7. *Suppose there is no legal pair on x. If there are maximal dangerous sets X, Y, Z and $u, v, w \in N(x)$ such that $u \in X \cap Y$, $v \in X \cap Z$, $w \in Y \cap Z$ and $u, v, w \notin X \cap Y \cap Z$, then either $d(x) = 3$ or there is a cut edge incident to x.*

Proof. We divide the proof into two cases.

Case 1: Inequality (2.3a) holds for at least one of $(X, Y), (X, Z), (Y, Z)$. Without loss of generality, assume inequality (2.3a) holds for (X, Y). Since $w \notin Y - X$, by the maximality of X, $s(X \cup Y) \geq 2$. By inequality (2.3a) for (X, Y), this implies that $s(X \cap Y) = d(X, Y) = 0$ and $s(X \cup Y) = 2$.

Consider $X \cap Y$ and Z. Suppose inequality (2.3a) holds for $(X \cap Y, Z)$, then $(X \cap Y) \cup Z$ will be dangerous, but this contradicts the maximality of Z since $u \in (X \cap Y) - Z$. Therefore, inequality (2.3b) must hold for $(X \cap Y, Z)$. Thus, $s(Z - (X \cap Y)) \leq s(X \cap Y) + s(Z) \leq 0 + 1 = 1$. Note that $Z - (X \cap Y)$ is non-empty since $v, w \in Z - (X \cap Y)$. This implies that $Z - (X \cap Y)$ is dangerous.

Define $Z' = Z - (X \cap Y)$; hence $X \cap Y \cap Z' = \emptyset$, see Figure 1(b). Consider $X \cup Y$ and Z'. Note that $\bar{d}(X \cup Y, Z') \geq d(v+w, x) \geq 2$ and $s(X \cup Y) + s(Z') \leq 2 + 1 = 3$. So inequality (2.3b) does not hold for $(X \cup Y, Z')$, and thus inequality (2.3a) must hold. Since $w \in Z' - X$, by the maximality of X, $X \cup Y \cup Z'$ cannot be dangerous, and hence $s(X \cup Y \cup Z') \geq 2$. By inequality (2.3a) for $(X \cup Y, Z')$, this implies that $s((X \cup Y) \cap Z') = s((X \cap Z') \cup (Y \cap Z')) \leq 1$. Note that $d(X, Y) = 0$ implies that $d(X \cap Z', Y \cap Z') = 0$. Applying the following claim with $S_1 := X \cap Z'$ and $S_2 := Y \cap Z'$ will show that either xv or xw is a cut edge, completing the proof of Case 1.

Claim 2.8. *For two disjoint vertex sets S_1, S_2 with x-neighbours $x_1 \in N(x) \cap S_1$, $x_2 \in N(x) \cap S_2$, if $d(S_1, S_2) = 0$ and $S_1 \cup S_2$ is dangerous, then there is a cut edge incident to x.*

Proof. Since $S_1 \cup S_2$ is dangerous, we have

$$\begin{aligned} 1 &\geq d(S_1 \cup S_2) - r(S_1 \cup S_2) \\ &\geq d(S_1) + d(S_2) - \max\{r(S_1), r(S_2)\} \\ &\geq \min\{d(S_1), d(S_2)\}. \end{aligned}$$

This implies $d(S_1) \leq 1$ (or $d(S_2) \leq 1$) and hence xx_1 (or xx_2) is a cut edge incident to x. □

Case 2: Inequality (2.3a) does not hold for any pair $(X, Y), (X, Z), (Y, Z)$. In other words, inequality (2.3b) holds in these three pairs. Consider X and Y; $\bar{d}(X, Y) \geq d(u, x) \geq 1$ and $s(X) + s(Y) \leq 1 + 1 = 2$. By inequality (2.3b) for (X, Y), this implies that $s(X - Y) = s(Y - X) = 0$. Consider $X - Y$ and Z, $\bar{d}(X - Y, Z) \geq d(v, x) \geq 1$ and $s(X - Y) + s(Z) \leq 0 + 1 = 1$, and so inequality (2.3b) does not hold for $(X - Y, Z)$. Thus inequality (2.3a) must hold for $(X - Y, Z)$, and so $s((X - Y) \cup Z) \leq s(X - Y) + s(Z) \leq 0 + 1 = 1$. Therefore, $(X - Y) \cup Z$ is dangerous. By the maximality of Z, $X - Y - Z$ must be empty. Using similar argument, $Y - X - Z$ and $Z - X - Y$ are also empty, see Figure 1(c). Since inequality (2.3b) holds for $(X, Y), (X, Z), (Y, Z)$ and X, Y, Z are all dangerous, $\bar{d}(X, Y) = d(u, x) = 1$, $\bar{d}(X, Z) = d(v, x) = 1$, $\bar{d}(Y, Z) = d(w, x) = 1$. Therefore $d(X \cup Y \cup Z, V - (X \cup Y \cup Z) - x) = 0$. Suppose $d(x) \neq 3$. Consider another x-neighbour t, then $t \in V - X \cup Y \cup Z$. Since $\bar{d}(X, Y) = 1$, $ut \notin E$ and so there exists a dangerous set D containing u and t for (xu, xt) to be illegal. Applying Claim (2.8) with $S_1 := D - (X \cup Y \cup Z)$ and $S_2 := D \cap (X \cup Y \cup Z)$ implies that there is a cut edge incident to x. Therefore, either $d(x) = 3$ or there is a cut edge incident to x. This completes the proof of Case 2, and thus Theorem 2.2. □

An alternate proof of Mader's theorem: Without the simplicity constraint, as long as $d(x) \neq 3$, a similar argument as in Lemma 2.6 can be used to construct the 3-dangerous-set configuration, and then Lemma 2.7 will imply Mader's theorem. The details are deferred to the full version.

3 Degree Bounded Network Design with Metric Costs

In this section we present the approximation algorithms for degree bounded network design problems with metric costs. Our algorithms can be seen as a generalization of Christofides' algorithm on metric TSP. The main technical tool is the simplicity preserving edge splitting-off operation, which is used to short-cut high degree vertices while maintaining connectivity requirements and preserving simplicity. The following is an overview of the algorithm for the case of local edge-connectivity.

First we use Jain's algorithm [19] to compute a Steiner network whose cost is no more than twice the optimal cost. Note that there may be vertices with degree larger than $r_{\max}$. We plan to use the simplicity preserving edge splitting-off operation to short-cut those vertices. To do so we need to make sure that the conditions in Theorem 2.2 are satisfied. If $r_{\max} = 1$, there is a simple 2-approximation algorithm for the minimum bounded degree Steiner network problem. Hence we assume $r_{\max} \geq 2$, and thus $d(v) \neq 3$ when $|N(v)| \geq r_{\max} + 2$. We also augment the Steiner network so that each connected component is 2-edge-connected, with a small increase in the cost (see Section 3.1.5), and thus there is no cut edge in the Steiner network. In Section 3.1.1, we show that if $|N(v)| \geq r_{\max} + 2$ and $N(v)$ is a clique, then we can remove redundant edges without violating any connectivity requirements and without introducing cut edges. With all the conditions satisfied, we can apply Theorem 2.2 on a vertex with $|N(v)| \geq r_{\max} + 2$. Call a vertex $u \in V$ *r-even* if $d(u)$ has the same parity as $r_{\max}$, and call u *r-odd* if $d(u)$ has different parity than $r_{\max}$. For every r-even vertex, by repeatedly applying Theorem 2.2, its degree can be reduced to at most $r_{\max}$. Similarly, for every r-odd vertex, its degree can be reduced to at most $r_{\max} + 1$ by repeatedly applying Theorem 2.2. Since the cost function satisfies triangle inequalities, the cost of the resulting Steiner network is no more than the cost of the initial Steiner network. This is an outline of the proof of the first part of Theorem 1.1 and Theorem 1.3.

For the second part of Theorem 1.1 and Theorem 1.3, we need to further reduce the maximum degree from $r_{\max}+1$ to $r_{\max}$. Assume for simplicity that $r_{\max}$ is even, and thus the number of r-odd vertices is even. We add a minimum cost perfect matching on r-odd vertices to make them r-even, and so all the vertices with degree larger than $r_{\max}$ are of degree $r_{\max}+2$. Note that parallel edges may be created when we add a matching. In Section 3.1.3, we prove that the simplicity-preserving edge splitting-off operation can be performed on those vertices with degree $r_{\max}+2$ to maintain connectivity and *restore simplicity* again, so that the resulting graph is simple and has maximum degree $r_{\max}$. Now let us present the details for different settings.

3.1 Edge Connectivity

3.1.1 Removing Redundant Edges

Suppose $|N(x)| \geq r_{\max}+2$ and $N(x)$ is a clique. We show that there are $u, v \in N(x)$ so that removing the edges uv, xu, xv maintains the local edge-connectivity of all pairs. To do so, it suffices to prove that $s(X) \geq 0$ for all $X \subset V$. Suppose a set $D \subset V$ with $d(D) < r(D)$ after removing the edges uv, xu, xv. By symmetry, assume $x \in D$. For $d(D) < r(D)$, at least one of $u, v \in V - D$. Let us assume $u \in V - D$. We have $d(D) \geq d(x, N(x) \cap (V-D)) + d(u, N(x) \cap D) = |N(x)| \geq r_{\max} \geq r(D)$, a contradiction. Therefore removing the edges uv, xu and xv will maintain the local edge-connectivity for all pairs. Also, there will be no cut edges introduced. Furthermore, the parities of the degrees of x, u, v remain the same. Henceforth, we assume that whenever $d(x) \geq r_{\max}+2$, then $N(x)$ is not a clique.

3.1.2 Perfect Matching

In the global edge-connectivity setting, we show that the cost of a minimum cost perfect matching between k-odd vertices is at most $EC_k(G)/k$, where $EC_k(G)$ denotes the optimal cost of a k-edge-connected subgraph and k-odd vertices are vertices with degree of different parity than k. Let the set of k-odd vertices be T. First we assume that $|T|$ is even. When the cost function satisfies triangle inequalities, the cost of a minimum cost perfect matching between T is equal to the cost of a minimum T-join, where a T-join is a subgraph in which T is equal to the set of vertices with odd degree. Let H be a k-edge-connected subgraph with minimum cost. Since H is k-edge-connected, by setting $x_e = 1/k$ for each edge $e \in H$, it is a feasible solution to the up hull of the T-join polytope [35]. Since the T-join polytope is integral, this implies that the cost of a minimum cost perfect matching between T is at most $EC_k(G)/k$.

When $|T|$ is odd, then k is odd, and there is no k-regular subgraph. Given a specific vertex v, if $v \in T$, we set $T' := T - \{v\}$; if $v \notin T$, we set $T' := T \cup \{v\}$. Then we can add a minimum cost perfect matching between T', so that v is the only vertex with degree $k+1$ in the resulting graph. By the same argument, the cost of this matching is at most $EC_k(G)/k$.

3.1.3 Edge Splitting-Off Restoring Simplicity

Suppose we are given a simple Steiner network with maximum degree $r_{\max}+1$. Suppose further that there is no cut edge and $r_{\max}$ is even. We need to further reduce its maximum degree to $r_{\max}$. First we add a minimum cost perfect matching between the r-odd vertices. Note that the resulting Steiner network may then have parallel edges. We plan to apply edge splitting-off operations to reduce the maximum degree to $r_{\max}$ and furthermore restore the simplicity of the Steiner network.

Consider a vertex x with degree $r_{\max}+2$. We can assume that $d(x) \neq 3$. If there are no parallel edges incident to x, then we can also assume that $N(x)$ is not a clique by Section 3.1.1, and thus we can apply Theorem 2.2 to reduce the degree of x to $r_{\max}$, without introducing new parallel edges. Now consider the case when there are parallel edges incident to x. Let v be the unique neighbour of x so that there are two parallel edges between x and v. If x and v have at least $r_{\max}$ common neighbours (which includes the case that $N(x)$ is a clique), then there are $r_{\max}$ edge-disjoint paths between x and v, and so both parallel edges between x and v can be removed while keeping local edge-connectivity requirement for all pairs. So assume that x and v have at most $r_{\max}-1$ common neighbours. If there exists u so that $xu \in E$ and $vu \notin E$ and xu, xv are admissible, then this is a simplicity-preserving edge splitting-off operation to reduce the degree of x to $r_{\max}$ and there is no more parallel edges incident to x. By repeatedly applying this operation, we can reduce the degree of every vertex to $r_{\max}$ while keeping connectivity requirements and restoring simplicity. It remains to prove that such an u must exist.

Suppose, by way of contradiction, that x has no neighbour u that (xu, xv) is a legal pair. Let x, v share $r_{\max}-l$ common neighbours with $l \geq 1$. Denote by $\{u_1, u_2, \ldots, u_l\}$ the set of neighbours of x that is not adjacent to v. Since xu_i, xv are not admissible for all u_i, there exists a dangerous set D_i such that $u_i, v \in D_i, x \notin D_i$ for $1 \leq i \leq l$. Since one parallel edge between xv is added in the matching, this implies that D_i is tight before the addition of the matching for all i. Consider the Steiner network G before the addition of the matching. Since $\bar{d}_G(D_i, D_j) \geq d(x, v) = 1$, inequality (2.3b) cannot hold for (D_i, D_j), and thus inequality (2.3a) must hold for (D_i, D_j). This implies that the union of these tight sets is tight in G. Therefore, there exists a tight set T in G such that $u_i, v \in T, x \notin T$ for $1 \leq i \leq l$, and thus $d_G(x, T) \geq l + 1$. In addition, the $r_{\max} - l$ common neighbours of x and v provide $r_{\max} - l$ edge-disjoint paths between x and v in G, Therefore, $d_G(T) \geq r_{\max} + 1$, which contradicts that T is a tight set in G. This shows that such a u must exist, and thus the simplicity-preserving edge splitting-off operation can be applied to obtain a simple Steiner network with maximum degree $r_{\max}$.

3.1.4 Proof of Theorem 1.1 and Theorem 1.2

Given any k-edge-connected graph with $k \geq 2$, after removing redundant edges as in Section 3.1.1, we can apply Theorem 2.2 repeatedly to obtain a simple k-edge-connected graph with maximum degree $k + 1$, without increasing the cost. This proves Theorem 1.1(1). As in Section 3.1.2, we can add a perfect matching between k-odd vertices with cost at most $EC_k(G)/k$. Then, as in Section 3.1.3, we can apply the simplicity-preserving edge splitting-off operation once again to obtain a simple k-edge-connected subgraph with maximum degree k, without increasing the cost. This proves Theorem 1.1(2). Finally Theorem 1.2 follows by using a 2-approximation algorithm to obtain a simple k-edge-connected subgraph [22] as the initial k-edge-connected subgraph.

3.1.5 Proof of Theorem 1.3 and Theorem 1.4

Suppose we are given a Steiner network H. In order to apply Theorem 2.2 to short-cut the high degree vertices, we first augment the Steiner network so that each connected component is 2-edge-connected, and thus there is no cut edge in the resulting Steiner network. One way to augment the graph is as follows: Double H to obtain H' so that each connected component of H is 2-edge-connected, then short-cut and remove redundant edges to make H' simple and each connected component 2-edge-connected. The cost of H' is at most twice the cost of H. We can then apply Theorem 2.2 to obtain a simple Steiner network which has maximum degree $r_{\max} + 1$, without increasing the cost. This proves Theorem 1.3(1). In the following we assume $r_{\max}$ is even. We add a minimum cost perfect matching on r-odd vertices in each component of the current Steiner network. Then we apply Theorem 2.2 once again to obtain a simple Steiner network with maximum degree $r_{\max}$ as in Section 3.1.3. Note that the cost of the matching is at most the cost of H, which can be proved by standard doubling and short-cutting argument. Therefore, the cost of the resulting Steiner network is at most 3 times the cost of the initial Steiner network H, which proves Theorem 1.3(2). Finally, Theorem 1.4 follows by using the Steiner network return by Jain's algorithm [19] as the initial Steiner network H, which has cost at most 2OPT.

3.2 Vertex Connectivity

We consider the minimum bounded degree k-vertex-connected subgraph problem. The algorithm is similar to that of the minimum bounded degree k-edge-connected subgraph problem, with some technical subtleties. Given any k-vertex-connected subgraph, the plan is to use the edge splitting-off operation to reduce the degree of all vertices to at most $k + 1$ while maintaining k-vertex-connectivity of the graph. Splitting-off operations maintaining vertex-connectivity is first studied by Bienstock, Brickell, Monma [6], where they prove the following theorem which implies Theorem 1.5(1).

Theorem 3.1 (Bienstock, Brickell, Monma [6]). *Let G be a minimally k-vertex-connected graph with $|V| \geq 2k$ and $|V| \geq 2$. If $x \in V$ has degree at least $k+2$, then either:*

1. *there is an edge splitting-off on x that maintains k-vertex-connectivity;*
2. *there is a vertex y such that for any edge splitting-off on x, there is an edge splitting-off on y such that both operations perform simultaneously would maintain k-vertex-connectivity.*

To prove the second part of Theorem 1.5, we use a similar strategy as in the edge-connectivity setting. In the following we assume the number of k-odd vertices to be even. We add a minimum cost perfect matching on the k-odd vertices. The argument in Section 3.1.2 implies the cost of the matching is at most $1/k$ the cost of an optimal k-edge-connected subgraph, which is at most $1/k$ the cost of an optimal k-vertex-connected subgraph. Then we plan to apply edge splitting-off again to decrease the maximum degree to k. However, after the matching is added, the graph is no longer *minimally* k-vertex-connected, and so Theorem 3.1 cannot be applied directly. Cheriyan, Jordán and Nutov [8] proved a similar theorem as Theorem 3.1 by removing the minimality assumption, but the degree is replaced by $k + 3$, which is not sufficient for our purpose. We strengthen Theorem 3.1 by removing the assumption that the graph is minimally k-vertex-connected. The proof is very similar to the proof of Theorem 3.1, which is deferred to the full version of this paper. Also, parallel edges may be created after a matching is added. Nevertheless, we can obtain a k-vertex-connected subgraph with maximum degree k, by applying edge splitting-off again as in Section 3.1.3, whose details are also deferred to the full version of this paper. This completes the outline of the proof of Theorem 1.5(2). Finally, Theorem 1.6 follows by using the algorithm of Kortsarz and Nutov [24] to find a k-vertex-connected subgraph as the initial graph, which has cost at most $2 + \frac{k-1}{n}$ times the optimal cost.

3.3 Spanning Trees

We present an approximation algorithm for the minimum bounded degree spanning tree problem with metric costs. Given a spanning tree T, denote by $\deg_T(v)$ the degree of a vertex v in the tree and $d(v)$ the degree bound for v. Using a minimum-cost flow technique, Fekete et.al. [12] showed that the cost of a tree satisfying degree bounds is at most the cost of the original tree times

$$2 - \min\left\{\frac{d(v)-2}{\deg_T(v)-2} : v \in V, \deg_T(v) > 2\right\}.$$

From the expression it can be seen that if $\deg_T(v)$ is closer to the degree bound $d(v)$, then the performance guarantee is better. Therefore one natural approach is to find a minimum spanning tree with smallest maximum degree. On 2-dimensional Euclidean plane, there is a minimum spanning tree of maximum degree 5 [34], and Khuller et.al. [22] showed how to convert such a spanning tree to a spanning tree with maximum degree 3 and 4 with cost no more than 1.5 and 1.25 times the minimum spanning tree respectively. Further improvements are in [7, 21], and there is a quasi-polynomial approximation scheme in [1]. Khuller et.al. [22] also showed in Euclidean space (not necessarily on plane), finding a spanning tree with maximum degree 3 is approximable within a factor of $5/3$.

For general metric space, it is not necessarily true that there is a minimum spanning tree with small maximum degree. However, on general weighted graphs, Singh and Lau [36] gave an algorithm to find a spanning tree with maximum degree $B+1$, whose cost is no more than the optimal cost of a spanning tree of maximum degree B. Therefore, we could first use the algorithm by Singh and Lau to obtain a spanning tree with degree violation at most 1, and then apply the minimum cost flow technique of Fekete et.al. to construct a spanning tree satisfying all the degree bounds. This implies Theorem 1.7, which improves upon the 2-approximation algorithm for general metric space.

Acknowledgment

We thank Samir Khuller for pointing us to the paper [12], and Tibor Jordán for discussions on edge splitting-off results.

References

[1] S. Arora, K.L. Chang: Approximations schemes for degree-restricted MST and red-blue separation problem. Algorithmica 40(3): 189-210, 2004.

[2] J. Bang-Jensen, A. Frank, B. Jackson: Preserving and increasing local edge-connectivity in mixed graph. SIAM J. Discrete Math. 8(2), 155-178, 1995.

[3] J. Bang-Jensen, T. Jordán: Edge-connectivity augmentation preserving simplicity. SIAM J. Discrete Math. 11(4): 603-623, 1998.

[4] J. Bang-Jensen, T. Jordán: Splitting off edges within a specified subset preserving the edge-connectivity of the graph. Journal of Algorithms 37(2), 326-343, 2000.

[5] A. Bernáth, T. Király: A new approach to splitting-off. Proceedings of 13th International Conference on Integer Programming and Combinatorial Optimization, 401-415, 2008.

[6] D. Bienstock, E.F. Brickell, C.L. Monma: On the structure of minimum-weight k-connected spanning networks. SIAM J. Discrete Math. 3(3) 320-329, 1990.

[7] T.M. Chan: Euclidean bounded-degree spanning tree ratios. Proc. 19th Symp. Computational Geometry, 11-19, 2003.

[8] J. Cheriyan, T. Jordán, Z. Nutov: On rooted node-connectivity problems. Algorithmica 30: 353-375, 2001.

[9] J. Cheriyan, A. Vetta: Approximation algorithms for network design with metric costs. SIAM J. Discrete Math. 21(3): 612-636, 2007.

[10] N. Christofides: Worst-case analysis of a new heuristic for the travelling salesman problem. Report 388, Graduate School of Industrial Administration, CMU, 1976.

[11] A. Czumaj, A. Lingas: Approximation schemes for minimum-cost k-connectivity problems in geometric graphs. Handbook of Approximation Algorithms and Metaheuristics, Ch. 51 (2007).

[12] S.P. Fekete, S. Khuller, M. Klemmstein, B. Raghavachari, N. Young: A network-flow technique for finding low-weight bounded-degree trees. Journal of Algorithms 24(2), 310-324, 1997.

[13] A. Frank: Augmenting graphs to meet edge-connectivity requirements, SIAM J. Discrete Math. 5(1): 25-53, 1992.

[14] A. Frank: Edge-connection of graphs, digraphs, and hypergraphs. EGRES - Egerváry Research Group on Combinatorial Optimization, Technical Report, TR-2001-11.

[15] T. Fukunaga, H. Nagamochi: Approximating a generalization of metric TSP. IEICE Trans. Inf. & Syst., VOL.E90D, 2007.

[16] T. Fukunaga: Approximation algorithms to the network design problems. Doctoral Dissertation 2007.

[17] T. Fukunaga, H. Nagamochi: Network design with edge connectivity and degree constraints. Approximation and Online Algorithms, Lecture Notes in Computer Science, Springer, Volume 4368, 188-201, 2007.

[18] M.X. Goemans: Minimum bounded-degree spanning trees. Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science, 273-282, 2006.

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

[20] T. Jordán: Constrained edge-splitting problems. SIAM J. Discrete Math. 17(1), 88-102, 2003.

[21] R. Jothi, B. Raghavachari: Degree-bounded minimum spanning trees. Proc. 16th Canadian Conf. on Computational Geometry: 192-195, 2004.

[22] S. Khuller, B. Raghavachari, N. Young: Low degree spanning trees of small weight. SIAM Journal on Computing 25, 355-368, 1994.

[23] S. Khuller, U. Vishkin: Biconnectivity approximations and graph carvings. Journal of the ACM 41, 214-235, 1994.

[24] G. Kortsarz, Z. Nutov: Approximating node connectivity problems via set covers. Algorithmica 37(2), 75-92, 2003.

[25] G. Kortsarz and Z. Nutov: Approximating min-cost connectivity problems, Survey Chapter in Approximation Algorithms and Metaheuristics, Editor T.F. Gonzalez, 2006.

[26] M. Kriesell: Edge-disjoint trees containing some given vertices in a graph. J. Combin. Theory B, 88, 53-63, 2003.

[27] L.C. Lau: An approximate max-Steiner-tree-packing min-Steiner-cut theorem. Combinatorica 27(1), 71-90, 2007.

[28] L.C. Lau, J. Naor, M.R. Salavatipour, M. Singh: Survivable network design with degree or order constraints. Proceedings of the 39th Annual Symposium on Theory of Computing, 651-660, 2007.

[29] L.C. Lau, M. Singh: Additive approximation for bounded degree survivable network design. Proceedings of the 40th Annual Symposium on Theory of Computing,759-768, 2008.

[30] L. Lovász: Combinatorial problems and exercises, North-Holland, 1993.

[31] W. Mader: A reduction method for edge-connectivity in graphs. Annals of Discrete Mathematics 3, 145-164, 1978.

[32] W. Mader: Konstruktion aller n-fach kantenzusammenhängenden Digraphen, Europ. J. Combinatorics 3, 63-67, 1982.

[33] C.L. Monma, B.S. Munson, W.R. Pulleyblank: Minimum-weight two-connected spanning networks. Math. Prog. 46, 153–171 (1990).

[34] C.L. Monma, S. Suri: Transitions in geometric minimum spanning trees (extended abstract). Proceedings of the seventh annual symposium on Computational geometry, 1991.

[35] A. Schrijver: Combinatorial optimization - polyhedra and efficiency. Springer-Verlag, Berlin, 2003.

[36] M. Singh, L.C. Lau: Approximating minimum bounded degree spanning trees to within one of optimal. Proceedings of the 39th Annual Symposium on Theory of Computing, 661-670, 2007.