

RALF BORNDÖRFER MARIKA KARBSTEIN

An Approximation Algorithm for the Steiner Connectivity Problem

Zuse Institute Berlin
Takustr. 7
14195 Berlin
Germany

Telephone: +49 30-84185-0
Telefax: +49 30-84185-125

E-mail: bibliothek@zib.de
URL: <http://www.zib.de>

ZIB-Report (Print) ISSN 1438-0064
ZIB-Report (Internet) ISSN 2192-7782

[§]Supported by the DFG Research Center MATHEON “Mathematics for key technologies”
*Zuse Institute Berlin, Takustr. 7, 14195 Berlin, Germany;
Email borndoerfer@zib.de, karbstein@zib.de

An Approximation Algorithm for the Steiner Connectivity Problem[§]

Ralf Borndörfer* Marika Karbstein*

Abstract

This paper presents an approximation algorithm for the Steiner connectivity problem, which is a generalization of the Steiner tree problem that involves paths instead of edges. The problem can also be seen as hypergraph-version of the Steiner tree problem; it comes up in line planning in public transport. We prove a $k + 1$ approximation guarantee, where k is the minimum of the maximum number of nodes in a path minus 1 and the maximum number of terminal nodes in a path. The result is based on a structural degree property for terminal nodes.

1 Introduction

The *Steiner connectivity problem* is to connect a subset of nodes, the *terminal nodes*, in a graph by a set of paths (out of a given set) with minimum cost. It can be seen (i) as a Steiner tree problem in hypergraphs by interpreting each path as a hyperedge in a hypergraph with the same node and terminal node set, (ii) as a generalization of the Steiner tree problem in graphs, where we consider a set of paths instead of edges, and (iii) as a prototype for all problems where nodes are connected by paths. A typical application is line planning with integrated passenger routing in public transport, see, e.g., [3, 10, 12]. A detailed investigation of and a survey on results about the Steiner connectivity problem can be found in [9].

The Steiner connectivity problem is \mathcal{NP} -hard in general and even when all nodes are terminal nodes. If the number of terminal nodes is fixed, it can be solved in (pseudo-) polynomial time by dynamic programming using a reduction to the directed Steiner tree problem, compare with [4] and [7]. The two terminal case even gives rise to a TDI formulation and associated min-max theorems [2]. The case when all nodes are terminals can be interpreted as a submodular set covering problem [2]. As such, it allows a best-possible approximation ratio of $\log(|V|)$, where V is the set of nodes [14]. Alternative combinatorial proofs based on polymatroids and a direct Chvátal-type argument [6] were given in [1] and [2], respectively. A polyhedral model hierarchy similar to the Steiner tree case [5, 11] has been derived [4], that yields very good computational results. This brings up the question whether an approximation algorithm with a guaranteed quality exists for the general case including Steiner nodes.

We give the first such algorithm extending the primal-dual method of Goemans and Williamson [8]. The algorithm yields a $(k + 1)$ -approximation guarantee for the Steiner connectivity problem where k is the minimum of (a) the maximum number of edges in a path and (b) the maximum number of terminal nodes in a path. Note that we have $k = 1$ for the Steiner tree problem, i. e., the well-known 2-approximation result of Goemans and Williamson [8] for this special case is included. The key to proving the approximation guarantee is a structural result that the average *path-degree* of a terminal node (number of paths incident to the node) in an inclusion wise minimal solution is at most $k + 1$.

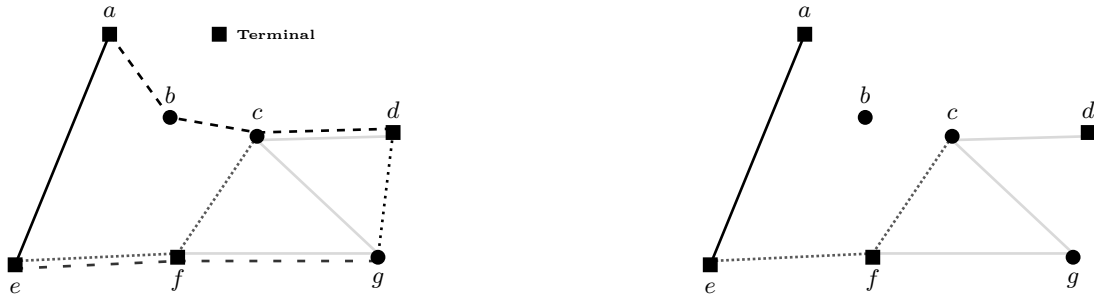


Figure 1: Example of a Steiner connectivity problem. *Left:* A graph with four terminal nodes ($T = \{a, d, e, f\}$) and six paths ($\mathcal{P} = \{p_1 = (a, b, c, d), p_2 = (e, f, g), p_3 = (a, e), p_4 = (e, f, c), p_5 = (g, d), p_6 = (f, g, c, d)\}$). *Right:* A feasible solution with three paths ($\mathcal{P}' = \{p_3, p_4, p_6\}$).

None of the arguments and results in our paper require a particular order of the nodes in a path. It is only relevant that the nodes in a path are connected. Therefore, paths could be replaced by hyperedges and all arguments can be interpreted in hypergraph terminology. Indeed, a slightly weaker $(k + 1)$ -approximation result for case (a) was stated in the context of hypergraphs by Takeshita, Fujito, and Watanabe [13] in a paper written in Japanese, however, without a proof. In particular, as far as we could find out, they do not give a proof for the degree property. Case (b) is new and extends the result. We remark that no transformation to the graphical case is known that would yield the degree property or any of the approximation results.

We use the following notation. We are given an undirected, connected graph $G = (V, E)$, a set of *terminal nodes* $T \subseteq V$, and a set of elementary *paths* \mathcal{P} in G . We denote by $V(p) \subseteq V$ the set of nodes and by $E(p) \subseteq E$ the set of edges of p ; $V(\mathcal{P}') = \cup_{p \in \mathcal{P}'} V(p)$ and $E(\mathcal{P}') = \cup_{p \in \mathcal{P}'} E(p)$, $\mathcal{P}' \subseteq \mathcal{P}$. We assume that each edge is covered by at least one path $p \in \mathcal{P}$; in particular, G has no loops. A set $\mathcal{P}' \subseteq \mathcal{P}$ is *T-connecting* if every two nodes in T are connected in the subgraph $H = (V, E(\mathcal{P}'))$. By assumption \mathcal{P} is V -connecting since $H = G$ is connected. The paths have nonnegative costs $c \in \mathbb{R}_+^{\mathcal{P}}$. The Steiner connectivity problem is to find a T -connecting set $\mathcal{P}' \subseteq \mathcal{P}$ of minimum cost. Figure 1 gives an example of a Steiner connectivity problem and a feasible solution. The *length* $|p| = |E(p)|$ of a path is the number of edges it contains. Finally, we denote by $\deg_{\mathcal{P}}(v) = |\{p \in \mathcal{P} : v \in V(p)\}|$ the *path-degree* w. r. t. \mathcal{P} of node $v \in V$. We skip “w. r. t. \mathcal{P} ” in the notation if there is no danger of confusion.

The article is structured as follows. We state the degree property and a self-contained proof in Section 2. In Section 3 we will describe and prove correctness and approximation ratio of the primal-dual algorithm for the Steiner connectivity problem.

2 The Degree Property

If the length of all paths in \mathcal{P} is 1, i. e., the paths correspond to edges, a minimal V -connecting set is a spanning tree in G . A tree has $|V|$ nodes and $|V| - 1$ edges, and each edge is incident to exactly two nodes. The average node degree in a tree is therefore $\frac{2(|V|-1)}{|V|} \leq 2 - \frac{2}{|V|} \leq 2$, i. e., at most 2. It is well known that this bound also holds for the average degree of a terminal node in an inclusion wise minimal Steiner tree, because each non-terminal node has degree at least 2. In contrast to that, a non-terminal node in an inclusion wise minimal T -connecting set can have a path-degree of 1, compare with node g in the right of Figure 1. However, the degree property of terminal nodes in Steiner trees can be generalized to minimal T -connecting sets as

Algorithm 1: Constructively increase a connecting set.

Input : A connected graph $G = (V, E)$, a set of terminal nodes $T \subseteq V$, an inclusion wise minimal T -connecting set \mathcal{P}' , a subset of terminal nodes $T_{i-1} \subset T$, an inclusion wise minimal T_{i-1} -connecting set \mathcal{P}_{i-1} ; $\mathcal{P}_{i-1} \subset \mathcal{P}'$

Output: An inclusion wise minimal T_i -connecting set \mathcal{P}_i , $T_{i-1} \subset T_i \subset T$, $\mathcal{P}_{i-1} \subset \mathcal{P}_i \subset \mathcal{P}'$.

Init : $\mathcal{P}_i := \mathcal{P}_{i-1}$, $T_i = T_{i-1}$

```

1 Choose  $p_i^1 \in \mathcal{P}' \setminus \mathcal{P}_{i-1}$  s.t.  $V(p_i) \cap (\cup_{p \in \mathcal{P}_{i-1}} V(p)) \neq \emptyset$ .
2  $\mathcal{P}_i := \mathcal{P}_{i-1} \cup \{p_i^1\}$ 
3  $T_i := T_{i-1} \cup (V(p_i^1) \cap T)$ 
4  $r_i := |T_i \setminus T_{i-1}|$ 
5  $\ell_i := 1$ 
6 while  $r_i = 0$  do
7    $\ell_i := \ell_i + 1$ 
8   Choose  $p_i^{\ell_i} \in \mathcal{P}' \setminus \mathcal{P}_i$  s.t.  $V(p_i^{\ell_i}) \cap T_i = \emptyset$ ,  $V(p_i^{\ell_i-1}) \cap V(p_i^{\ell_i}) \neq \emptyset$  and
      $V(p_i^{\ell_i}) \cap (\cup_{q \in \mathcal{P}_{i-1}} V(q)) = \emptyset$ .
9    $\mathcal{P}_i := \mathcal{P}_i \cup \{p_i^{\ell_i}\}$ 
10   $T_i := T_i \cup (V(p_i^{\ell_i}) \cap T)$ 
11   $r_i := |T_i \setminus T_{i-1}|$ 
12 end

```

follows.

Lemma 2.1 (Degree Lemma). *The average path-degree of a terminal node w. r. t. an inclusion wise minimal T -connecting set \mathcal{P}' is at most $(k + 1)$, where k denotes the minimum of*

- (a) *the maximum number of edges in a path,*
- (b) *the maximum number of terminal nodes in a path.*

More precisely, we have

$$(a) \sum_{t \in T} \deg_{\mathcal{P}'}(t) \leq (k + 1)(|T| - k)$$

$$(b) \sum_{t \in T} \deg_{\mathcal{P}'}(t) \leq \begin{cases} (k + 1)(|T| - (k - 1)) & \text{if } k \geq 2 \\ (k + 1)(|T| - 1) & \text{if } k = 1, (|T| > 1) \end{cases}$$

$$k = \min\{\max_{p \in \mathcal{P}} |p|, \max_{p \in \mathcal{P}} |T \cap V(p)|\}.$$

Proof. We prove the claim by induction. To this purpose we consider an inclusion-wise minimal T -connecting set \mathcal{P}' . We first prove the claim for a T_0 -connecting set \mathcal{P}_0 with $T_0 \subseteq T$ and $\mathcal{P}_0 \subseteq \mathcal{P}'$. Then we constructively increase T_0 and \mathcal{P}_0 and show that, if the claim holds for T_{i-1} -connecting set \mathcal{P}_{i-1} with

$$T_0 \subset T_1 \subset \dots \subset T_{i-1} \dots \subset T \text{ and } \mathcal{P}_0 \subset \mathcal{P}_1 \subset \dots \subset \mathcal{P}_{i-1} \dots \subset \mathcal{P}',$$

then it also holds for T_i -connecting set \mathcal{P}_i , $T_{i-1} \subset T_i \subseteq T$, $\mathcal{P}_{i-1} \subset \mathcal{P}_i \subseteq \mathcal{P}'$.

Consider Algorithm 1. The algorithm increases an inclusion-wise minimal T_{i-1} connecting set \mathcal{P}_{i-1} to a T_i -connecting set \mathcal{P}_i by adding the paths $p_i^1, \dots, p_i^{\ell_i-1}$ to \mathcal{P}_{i-1} , $\ell_i \geq 1$. The following holds:

1. If $\ell_i = 1$ then the path p_i^1 adds $1 \leq r_i \leq k$ new terminal nodes not contained in T_{i-1} and contains at most
 - (i) $k + 1 - r_i$ terminal nodes of the set T_{i-1} in case (a),
 - (ii) $k - r_i$ terminal nodes of the set T_{i-1} in case (b),
 i.e., the overall path degree increases by at most $r_i + k$.
2. If $\ell_i \geq 2$ the set of paths have the following characteristics (due to the construction in the algorithm).
 - (i) The path p_i^1 contains at most k terminal nodes of the set T_{i-1} .
 - (ii) The paths $p_i^2, \dots, p_i^{\ell_i-1}$ contain no terminal node. If one of these paths contains a terminal node not contained in $|T_{i-1}|$, then the iteration would have stopped earlier. If one of these paths contains a terminal node of the set T_{i-1} then the path p_i^1 would be redundant which contradicts the assumption that \mathcal{P}' is inclusion wise minimal for T .
 - (iii) The path $p_i^{\ell_i}$ contains at least $r_i \geq 1$ new terminal nodes and no node of the set $\cup_{q \in \mathcal{P}_{i-1}} V(q)$. Again, this follows since \mathcal{P}' is inclusion wise minimal

Hence the overall path degree increases by at most $r_i + k$.

Case (a): Let $p_0 \in \mathcal{P}'$ be a path with $k+1$ terminal nodes. If no such path exists, we have the situation of case (b) considered below. Let $T_0 := T \cap V(p_0)$ then p_0 is an inclusion wise minimal T_0 -connecting set. The overall degree of terminal nodes is $|T_0| = k + 1 = (|T_0| - k)(k + 1)$.

Assume the claim holds for the inclusion-wise minimal T_{i-1} -connecting set \mathcal{P}_{i-1} . We use Algorithm 1 to increase T_{i-1} and \mathcal{P}_{i-1} to an inclusion-wise minimal T_i -connecting set \mathcal{P}_i . Either the overall degree on terminal nodes is increased by at most $r_i + k + 1 - r_i \leq r_i + k$, if $\ell_i = 1$ or it is increased by k (contributed by p_i^1) + r_i (contributed by $p_i^{\ell_i}$), if $\ell_i \geq 2$. Since $|T_i| = |T_{i-1} + r_i|$, the overall degree on the terminal node set T_i can be bounded as follows

$$\begin{aligned} \sum_{t \in T_i} \deg_{\mathcal{P}_i}(t) &\leq \sum_{t \in T_{i-1}} \deg_{\mathcal{P}_{i-1}}(t) + k + r_i \leq (|T_i| - r_i - k)(k + 1) + k + r_i \\ &= (|T_i| - k)(k + 1) - r_i k - r_i + k + r_i \leq (|T_i| - k)(k + 1). \end{aligned}$$

Case (b). Consider at first the case $k = 1$. All paths contain at most one terminal node. There exists a set of paths $p_0^1, \dots, p_0^{\ell_0}$, $2 \leq \ell_0$, such that the graph induced by the paths is connected and only p_0^1 and $p_0^{\ell_0}$ contain two distinct terminal nodes $\{t_1, t_2\} =: T_0$. We have an inclusion wise minimal T_0 -connecting set and the overall degree on terminal nodes is $|T_0| = 2 = k + 1 = (|T_0| - 1)(k + 1)$. The rest follows analogously to case (a).

Now consider the case $k \geq 2$. Let $p_0 \in \mathcal{P}'$ be a path with k terminal nodes. Let $T_0 := T \cap V(p_0)$, then p_0 is an inclusion wise minimal T_0 -connecting set with overall degree on terminal nodes $|T_0| = k < (|T_0| - (k - 1))(k + 1)$.

Assume the claim holds for the inclusion-wise minimal T_{i-1} -connecting set \mathcal{P}_{i-1} . We use Algorithm 1 to increase T_{i-1} and \mathcal{P}_{i-1} to an inclusion-wise minimal T_i -connecting set \mathcal{P}_i . As in case (a) the overall degree on terminal nodes increases by at most $r_i + k$ and we get

$$\begin{aligned} \sum_{t \in T_i} \deg_{\mathcal{P}_i}(t) &\leq \sum_{t \in T_{i-1}} \deg_{\mathcal{P}_{i-1}}(t) + k + r_i \leq (|T_i| - r_i - (k - 1))(k + 1) + k + r_i \\ &= (|T_i| - (k - 1))(k + 1) - r_i k - r_i + k + r_i \\ &\leq (|T_i| - (k - 1))(k + 1). \end{aligned}$$

□

We briefly show that this bound is tight for case (a). Lemma 2.1. Consider the instance in the left of Figure 2. All nodes are terminal nodes. We have n nodes in the rim and k nodes in

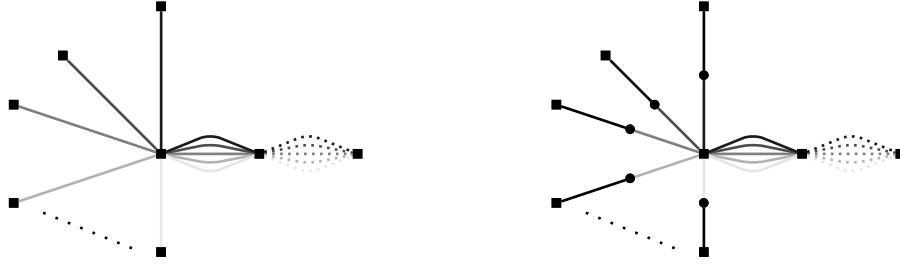


Figure 2: Example that the Degree Lemma 2.1 is tight for case (a) (*left*) and asymptotically tight for case (b) (*right*).

the middle. Suppose each path contains one node of the rim and all nodes in the middle. All paths together form a minimal V -connecting set. We have n nodes with path-degree 1 and k nodes with path-degree n , i. e., $|T| = n + k$ and the total degree is $n(k + 1) = (n + k - k)(k + 1)$, which gives an asymptotically average path-degree $(k + 1)$ as n goes to infinity. This instance can be slightly modified to get an asymptotically tight example for case (b), compare with the right of Figure 2. We have n additional non-terminal nodes in the inner rim and each path contains a non-terminal node in the inner rim and all nodes in the middle. We further have n paths that connect the outer rim with the inner rim. The maximum number of terminal nodes in a path is k and we have the same path degree for each terminal node as for the case (a) above.

3 The Primal-Dual Algorithm

In the following we will construct a primal-dual algorithm to find a T -connecting set. It generalizes the algorithm of Goemans and Williamson [8] for the Steiner forest problem. Our application to the Steiner connectivity problem is listed in Algorithm 2.

The algorithm constructs a T -connecting set \mathcal{P}' . In the beginning $\mathcal{P}' = \emptyset$ and each terminal node is considered to form a connected component that contains only itself. The idea is to extend and merge the connected components along paths until we have only one connected component left. In each iteration *moats* around the connected components are grown until a path *goes tight*. The radii of the moats correspond to the values of the dual variables for the cuts around the connected components, and a *path goes tight* if its associated inequality in the dual program becomes an equality, compare with equation (1).

More precisely, let B^i be the set of all connected components in iteration i of Algorithm 2; the initial set is $B^0 = \{\{t\} : t \in T\}$, i. e., the set of all terminal nodes. We iterate as long as B^i consists of more than one connected component. Denote by $B_p^i = \{b \in B^i : p \in \mathcal{P}_{\delta(b)}\}$ the set of connected components the path p “cuts” in iteration i . Here, $\mathcal{P}_{\delta(b)} := \{p \in \mathcal{P} : \delta(b) \cap E(p) \neq \emptyset\}$. We set $|B_p^i| = 0$ for $B_p^i = \emptyset$. In iteration i we choose, among paths $p \in \mathcal{P}$ with $|B_p^i| > 0$, a path for which the quotient of reduced cost and number of connected components the path cuts in iteration i is minimum, line 4. Denote by a^i this minimum value, line 5; it gives the maximum amount the dual variables can be increased. The associated path is added to \mathcal{P}' (ties broken arbitrarily), line 6, and the dual variables or moat radii are increased by the value a^i , line 8. If the path contains several connected components, these are merged into one connected component, line 10. All non-terminal nodes of the path are also added to the new connected component. Note that the path contains at least two connected components or at least one connected component and one non-terminal node. Line 12 is an updating step to prepare the computation of the next amount of increase of the dual variables. The final set of chosen paths

Algorithm 2: A primal-dual heuristic for the Steiner connectivity problem.

Input : A connected graph $G = (V, E)$, a set of paths \mathcal{P} with costs $c \in \mathbb{R}_{\geq 0}^{\mathcal{P}}$, a set of terminal nodes $T \subseteq V$.

Output: A T -connecting set $\mathcal{P}' \subseteq \mathcal{P}$.

- 1 $B^0 := \{\{t\} \mid t \in T\}$, $\mathcal{P}' := \emptyset$, $c_p^0 := c_p$ for all $p \in \mathcal{P}$, $i := 0$
- 2 $y_W := 0 \forall W \subset V$, $W \cap T \neq \emptyset$, $V \setminus W \cap T \neq \emptyset$ //only set when needed
- 3 **while** $|B^i| > 1$ **do**
- 4 $p = \operatorname{argmin}_{q \in \mathcal{P}} \left\{ \frac{c_q^i}{|B_q^i|} : |B_q^i| > 0 \right\}$
- 5 $a^i := \frac{c_p^i}{|B_p^i|}$
- 6 $\mathcal{P}' := \mathcal{P}' \cup \{p\}$
- 7 **for** all $b \in B^i$ **do**
- 8 $y_b := y_b + a^i$
- 9 **end**
- 10 $B^{i+1} := (B^i \setminus B_p^i) \cup \{b_1 \cup \dots \cup b_k \cup V(p)\}$ with $\{b_1, \dots, b_k\} := B_p^i$
- 11 **for** all $q \in \mathcal{P} \setminus \mathcal{P}'$ **do**
- 12 $c_q^{i+1} := c_q^i - |B_q^i| a^i$
- 13 **end**
- 14 $i := i + 1$
- 15 **end**
- 16 **for** all $p \in \mathcal{P}'$ **do**
- 17 **if** $\mathcal{P}' \setminus p$ is T -connecting **then**
- 18 $\mathcal{P}' := \mathcal{P}' \setminus p$ //deleting step
- 19 **end**
- 20 **end**

is T -connecting. In the end of the algorithm, lines 16 to 20, we consecutively remove paths as long as the resulting set is still T -connecting to obtain a minimal T -connecting set \mathcal{P}' .

Let $\mathcal{W} = \{W \subset V; \emptyset \neq W \cap T \neq T\}$. We call the set $\mathcal{P}_{\delta(W)} := \{p \in \mathcal{P} : \delta(W) \cap E(p) \neq \emptyset\}$ of all paths that cross the cut $\delta(W) = \{e \in E \mid |e \cap W| = 1\}$ at least once a *Steiner path cut*. The analysis of Algorithm 2 is based on the consideration of the following dual programs:

$$\begin{array}{ll}
 \min & \sum_{p \in \mathcal{P}} c_p x_p \\
 \text{s.t.} & \sum_{p \in \mathcal{P}_{\delta(W)}} x_p \geq 1 \quad \forall W \in \mathcal{W} \\
 & x_p \geq 0 \quad \forall p \in \mathcal{P}
 \end{array}
 \qquad
 \begin{array}{ll}
 \max & \sum_{W \in \mathcal{W}} y_W \\
 \text{s.t.} & \sum_{W \in \mathcal{W} : p \in \mathcal{P}_{\delta(W)}} y_W \leq c_p \quad \forall p \in \mathcal{P} \\
 & y_W \geq 0 \quad \forall W \in \mathcal{W}.
 \end{array}
 \tag{1}$$

The primal program is the LP relaxation of the undirected cut formulation of the Steiner connectivity problem. It minimizes the cost of a set of paths. The set covering inequalities guarantee T -connectedness. The dual program searches for a maximum cardinality packing of T -disconnecting node sets. The capacity constraints ensure that the node sets intersect each path at most c_p times, compare also with [4] and [9].

Proposition 3.1. *Setting $x_p = 1$ for all $p \in \mathcal{P}'$, $x_p = 0$ for all $p \in \mathcal{P} \setminus \mathcal{P}'$, and using variables y as defined at the end of Algorithm 2 gives solutions for the primal and dual programs (1).*

Proof. It is easy to see that \mathcal{P}' is a T -connecting set. Now, consider y_W , $W \in \mathcal{W}$. Clearly, $y_W \geq 0$, $W \in \mathcal{W}$. Let $p \in \mathcal{P}$ and r be the last iteration of the while-loop, i. e., at the end of this

last iteration a^r and c_p^{r+1} , $p \in \mathcal{P} \setminus \mathcal{P}'$, are defined. Then we get

$$\sum_{W \in \mathcal{W}: p \in \mathcal{P}_\delta(W)} y_W \stackrel{(i)}{=} \sum_{i=0}^r \sum_{b \in B_p^i} a^i \stackrel{(ii)}{=} c_p^0 - c_p^{r+1} \stackrel{(iii)}{\leq} c_p. \quad (2)$$

- (i) Compare with line 8 in Algorithm 2.
- (ii) We use the following equations, compare with line 12 in the algorithm,

$$c_p^{r+1} = c_p^r - \sum_{b \in B_p^r} a^r = c_p^0 - \sum_{i=0}^r \sum_{b \in B_p^i} a^i.$$

- (iii) This follows since $0 \leq c_p^{r+1} \leq c_p^0 = c_p$, compare with lines 1, 4, and 5 in Algorithm 2. Hence, y is feasible for the dual program in (1). Moreover, we have equality in (2) for $p \in \mathcal{P}'$, i. e.,

$$\sum_{W \in \mathcal{W}: p \in \mathcal{P}_\delta(W)} y_W = c_p. \quad (3)$$

□

Proposition 3.2. *Given a Steiner connectivity instance, let \mathcal{P}_{opt} be the minimum cost T -connecting set and \mathcal{P}' a T -connecting set computed with Algorithm 2. Then*

$$c(\mathcal{P}') \leq (k+1)c(\mathcal{P}_{\text{opt}}) \left(1 - \frac{k-1}{|T|}\right),$$

- i. e., Algorithm 2 is a $(k+1)$ -approximation algorithm with k being the minimum of
- (a) the maximum number of edges in a path,
 - (b) the maximum number of terminal nodes in a path.

Proof. Summing up the cost of all paths in \mathcal{P}' , we get

$$\sum_{p \in \mathcal{P}'} c_p \stackrel{(3)}{=} \sum_{p \in \mathcal{P}'} \sum_{W \in \mathcal{W}: p \in \mathcal{P}_\delta(W)} y_W = \sum_{W \in \mathcal{W}} \sum_{p \in \mathcal{P}_\delta(W) \cap \mathcal{P}'} y_W = \sum_{W \in \mathcal{W}} \deg_{\mathcal{P}'}(W) y_W.$$

If we can show the following

$$\sum_{W \in \mathcal{W}} \deg_{\mathcal{P}'}(W) y_W \leq (k+1) \left(1 - \frac{1}{|T|}\right) \sum_{W \in \mathcal{W}} y_W, \quad (4)$$

we are done. Note that this is more general than to require $\deg_{\mathcal{P}'}(W) \leq (k+1) \left(1 - \frac{1}{|T|}\right)$. We show inequality (4) by induction over the iterations of the algorithm. Initially, $y_W = 0$ for all $W \in \mathcal{W}$, so inequality (4) is true. Now, we have to show that the increase on the left hand side is smaller than the increase on the right hand side in every iteration, i. e., for each iteration i we have to show the following

$$\begin{aligned} a^i \sum_{b \in B^i} \deg_{\mathcal{P}'}(b) &\leq (k+1) \left(1 - \frac{k-1}{|B^i|}\right) \sum_{b \in B^i} a^i &\leq (k+1) \left(1 - \frac{k-1}{|B^i|}\right) \sum_{b \in B^i} a^i \\ \Leftrightarrow \sum_{b \in B^i} \deg_{\mathcal{P}'}(b) &\leq (k+1) (|B^i| - (k-1)) &= (k+1) \left(1 - \frac{k-1}{|B^i|}\right) |B^i| \\ & &\stackrel{|B^i| \leq |T|}{\leq} (k+1) \left(1 - \frac{k-1}{|T|}\right) |T|. \end{aligned}$$

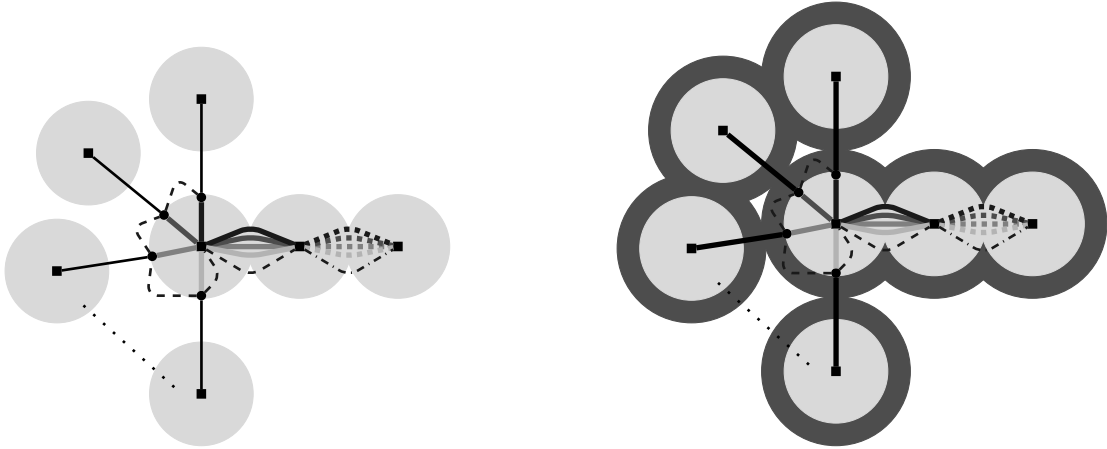


Figure 3: Example that the approximation guarantee of Algorithm 2 is asymptotically tight. *Left:* Iteration 0 to $n - 1$ (where n is number of nodes in the rim), *right:* Iteration n to $2n - 1$. The radii of the moats correspond to the values of the dual variables a , the chosen paths are drawn in thick lines.

Each $b \in B^i$ is connected. Consider the graph \tilde{G} that contains a node for each $b \in B^i$ and all nodes $v \in V$ that are not contained in one of the $b \in B^i$. Then the final set \mathcal{P}' restricted to \tilde{G} is a minimal B^i -connecting set. Suppose \mathcal{P}' is not minimal. Then there exists a path p that can be dropped. Since \mathcal{P}' is an inclusion wise minimal T -connecting set, there exists a component $b \in B^i$ that would not be connected without p . However, this contradicts the course of the algorithm in which the component $b \in B^i$ was connected by paths added before iteration i , i.e., by paths different from p . The rest follows with the Degree Lemma 2.1 since \mathcal{P}' is minimal. \square

Remark 3.3. Proposition 3.2 can be sharpened for case (a) and for $k = 1$ to

$$c(\mathcal{P}') \leq (k + 1) c(\mathcal{P}_{\text{opt}}) \left(1 - \frac{k}{|T|}\right) \quad \text{and} \quad c(\mathcal{P}') \leq 2 c(\mathcal{P}_{\text{opt}}) \left(1 - \frac{1}{|T|}\right),$$

respectively.

Algorithm 2 is polynomial. A naive implementation leads to a running time of $\mathcal{O}(|V||T| + k|V||\mathcal{P}|)$ for lines 1 to 15 and $\mathcal{O}(|V||\mathcal{P}|^2 + |E||\mathcal{P}|)$ for lines 16 to 20 which is dominated by $\mathcal{O}(|V||\mathcal{P}|^2)$ if $|\mathcal{P}| \geq |V|$.

Figure 3 illustrates the course of Algorithm 2 and shows that the bound for the approximation quality is asymptotically tight. There are k terminal nodes in the middle and n terminal nodes in the rim, n paths of cost k containing all terminal nodes in the middle plus 1 Steiner node in the rim, n paths of cost $1 + \epsilon$ containing 1 terminal node and one Steiner node in the rim, and 1 path of cost $k + \epsilon$ containing all terminal nodes in the middle and all Steiner nodes in the rim. In the first iteration, $a^0 = 1$ and one of the paths of cost k is chosen. The reduced cost are then 0 for the paths of original cost k , and ϵ for all other paths. The algorithm chooses all paths of reduced cost 0. In the n -th iteration, $a^n = \frac{\epsilon}{2}$ and the one-edge-paths are chosen. The path with original cost $k + \epsilon$ is not chosen because its reduced cost ratio is ϵ . The resulting solution has total cost of $(|T| - k)(k + 1 + \epsilon)$. The optimal solution consists of the paths with cost $k + \epsilon$ and all edge-paths, it has cost $(|T| - k)(1 + \epsilon) + k + \epsilon$. The ratio of the costs is asymptotically arbitrarily close to $k + 1$.

References

- [1] Gregor Baudis, Clemens Gröpl, Stefan Hougardy, Till Nierhoff, and Hans Jürgen Prömel. Approximating minimum spanning sets in hypergraphs and polymatroids. Technical report, HU Berlin, 2000.
- [2] Ralf Borndörfer, Nam Dung Hoang, Marika Karbstein, Thorsten Koch, and Alexander Martin. How many Steiner terminals can you connect in 20 years? In Michael Jünger and Gerhard Reinelt, editors, *Facets of Combinatorial Optimization; Festschrift for Martin Grötschel*, pages 215 – 244. Springer, 2013. ZIB Report 13-57.
- [3] Ralf Borndörfer and Marika Karbstein. A Direct Connection Approach to Integrated Line Planning and Passenger Routing. In Daniel Delling and Leo Liberti, editors, *ATMOS 2012 - 12th Workshop on Algorithmic Approaches for Transportation Modeling, Optimization, and Systems*, pages 47–57, Dagstuhl, Germany, 2012. Schloss Dagstuhl–Leibniz-Zentrum für Informatik.
- [4] Ralf Borndörfer, Marika Karbstein, and Marc E. Pfetsch. The Steiner connectivity problem. *Mathematical Programming*, 142(1):133–167, 2013.
- [5] Sunil Chopra and Mendu R. Rao. The Steiner tree problem I: Formulations, compositions and extension of facets. *Mathematical Programming*, 64(2):209–229, 1994.
- [6] Vašek Chvátal. A greedy heuristic for the set-covering problem. *Mathematics of Operations Research*, 4(3):233–235, 1979.
- [7] Jon Feldman and Matthias Ruhl. The directed steiner network problem is tractable for a constant number of terminals. In *IEEE Symposium on Foundations of Computer Science*, pages 299–308, 1999.
- [8] Michel X. Goemans and David P. Williamson. A general approximation technique for constrained forest problems. *SIAM Journal on Computing*, 24(2):296–317, 1995.
- [9] Marika Karbstein. *Line Planning and Connectivity*. PhD thesis, TU Berlin, 2013.
- [10] Karl Nachtigall and Karl Jerosch. Simultaneous network line planning and traffic assignment. In Matteo Fischetti and Peter Widmayer, editors, *ATMOS 2008 - 8th Workshop on Algorithmic Approaches for Transportation Modeling, Optimization, and Systems*, Dagstuhl, Germany, 2008. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, Germany.
- [11] Tobias Polzin. *Algorithms for the Steiner Problems in Networks*. PhD thesis, University of Saarland, Saarbrücken, 2003.
- [12] Anita Schöbel and Susanne Scholl. Line planning with minimal traveling time. In Leo G. Kroon and Rolf H. Möhring, editors, *Proceedings of 5th Workshop on Algorithmic Methods and Models for Optimization of Railways*, 2006.
- [13] Kazuhiko Takeshita, Toshihiro Fujito, and Toshimasa Watanabe. On primal-dual approximation algorithms for several hypergraph problems. *Joho Shori Gakkai Kenkyu Hokoku*, 99(15):13–18, 1999.
- [14] Laurence A. Wolsey. An analysis of the greedy algorithm for the submodular set covering problem. *Combinatorica*, 2(4):385–393, 1982.