Konrad-Zuse-Zentrum
für Informationstechnik Berlin

Takustraße 7
D-14195 Berlin-Dahlem
Germany

RALF BORNDÖRFER MARIKA KARBSTEIN

A Primal-Dual Approximation Algorithm for the Steiner Connectivity Problem

Supported by the DFG Research Center MATHEON "Mathematics for key technologies", project B15.

ZIB-Report 13-54 (September 2013)

Herausgegeben vom
Konrad-Zuse-Zentrum für Informationstechnik Berlin
Takustraße 7
D-14195 Berlin-Dahlem

Telefon: 030-84185-0
Telefax: 030-84185-125

e-mail: bibliothek@zib.de
URL: http://www.zib.de

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

A Primal-Dual Approximation Algorithm for the Steiner Connectivity Problem[§]

Ralf Borndörfer[*] Marika Karbstein[*]

Abstract

We extend the primal-dual approximation technique of Goemans and Williamson to the Steiner connectivity problem, a kind of Steiner tree problem in hypergraphs. This yields a $(k+1)$-approximation algorithm for the case that k is the minimum of the maximal number of nodes in a hyperedge minus 1 and the maximal number of terminal nodes in a hyperedge. These results require the proof of a degree property for terminal nodes in hypergraphs which generalizes the well-known graph property that the average degree of terminal nodes in Steiner trees is at most 2.

1 Introduction

Goemans and Williamson [3] developed a primal-dual approximation technique for graph problems. They showed that this technique yields a 2-approximation algorithm for Steiner tree and Steiner forest problems.

In this article we extend the primal-dual algorithm to the Steiner connectivity problem. This problem can be seen as a Steiner tree problem in hypergraphs. It can also be seen as a generalization of the Steiner tree problem in graphs, where we consider a set of paths instead of edges. More precisely, the task is to connect a subset of nodes, the *terminal nodes*, by a set of paths (out of a given set) with minimum cost. The Steiner connectivity problem is the prototype problem for line planning in public transport with integrated passenger routing which is considered in, e. g., [1, 6, 7]. A detailed investigation of the Steiner connectivity problem can be found in [2] and [4]. We prove that the primal-dual 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 2-approximation result for this special case is included.

Goemans and Williamson [3] prove the approximation factor for the Steiner tree problem by using the fact that the average degree of a terminal node (number of edges

[§]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

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

incident to the node) in a tree is at most 2. We generalize this result to our case. Namely, we show that the average *path-degree* of a terminal node (number of paths incident to the node) for an inclusion wise minimal solution is at most $k + 1$ where k is defined as in the above cases (a) and (b).

The Steiner connectivity problem can also be defined as a kind of Steiner tree problem in hypergraphs by interpreting each path as a hyperedge in a hypergraph with the same node and terminal node set. The problem is then to find a cost minimal set of hyperedges that connect all terminal nodes. The degree property can be interpreted in a straight forward way in hypergraphs.

The $(k+1)$-approximation result for case (a) was stated in the context of hypergraphs by Takeshita, Fujito, and Watanabe [5] in a paper written in Japanese. As far as we could find out, however, they do not give a proof for the degree property for this case. In fact, an inquiry with the authors and several other persons in the hypergraph community revealed that there is none published. Case (b) is new and extends the result.

We use the following notation. We are given an undirected 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 if $H = G$ 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 proof the primal-dual approximation 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 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 maximal number of edges in a path,*
- *(b) the maximal number of terminal nodes in a path.*

More precisely, we have

$$\sum_{t \in T} \deg_{\mathcal{P}'}(t) \leq (k+1)(|T|-1), \qquad k = \min\{\max_{p \in \mathcal{P}} |p|, \max_{p \in \mathcal{P}} |T \cap V(p)|\}.$$

Proof. We only consider paths that contain at least one terminal node since these are the only paths that contribute to the path-degree of the terminal nodes. Denote the set of these paths by $\mathcal{P}'(T)$. The idea of the proof is to consider these paths in such a sequence that either each path or a pair of paths establishes a connection to some new terminal. Reaching all terminals then requires at most $|T|-1$ such paths or pairs of paths. This gives rise to a sum of path-degrees at the terminal nodes of at most $(|T|-1)(k+1)$. The details of this argument are as follows. Define a starting order on $\mathcal{P}'(T)$ as follows.

$$\mathcal{P}'(T) = \{p_1, \ldots, p_n\} = \{p_1^1, \ldots, p_{s_1}^1, \ldots, p_1^\ell, \ldots, p_{s_\ell}^\ell\},$$

where
- ∘ $V(p_i^j) \cap (\cup_{r=1}^{i-1} V(p_r^j)) \neq \emptyset$, $j = 1, \ldots, \ell$, $i = 2, \ldots, s_j$, and
- ∘ $p_i^j \cap p_{\tilde{i}}^{\tilde{j}} = \emptyset$, for all $j \neq \tilde{j}$, $i = 1, \ldots, s_j$, $\tilde{i} = 1, \ldots, s_{\tilde{j}}$,

i.e., the graph induced by the paths $p_1^j, \ldots, p_i^j$, $i \leq s_j$, $j \in \{1, \ldots, \ell\}$, is connected and there is no connection to the graph induced by the paths $p_1^{\tilde{j}}, \ldots, p_{\tilde{i}}^{\tilde{j}}$, $\tilde{i} \leq s_{\tilde{j}}$, $\tilde{j} \in \{1, \ldots, \ell\}$, $j \neq \tilde{j}$.

We define $T_i = \cup_{j=1}^{i} (V(p_i) \cap T)$ to be the set of terminal nodes that are covered by $p_1, \ldots, p_i$. We have $T_i \subseteq T_{i+1}$, $i = 1, \ldots, n-1$, and $T_i = T_{i+1}$ is also possible. Figure 2 shows the notation of this proof on an example.

Let $r_1 \geq 1$ be the number of terminal nodes contained in path p_1, i.e., $r_1 = |T_1|$. For $i \geq 2$ let $r_i = |T_i \setminus T_{i-1}|$ be the number of additional terminal nodes contained in

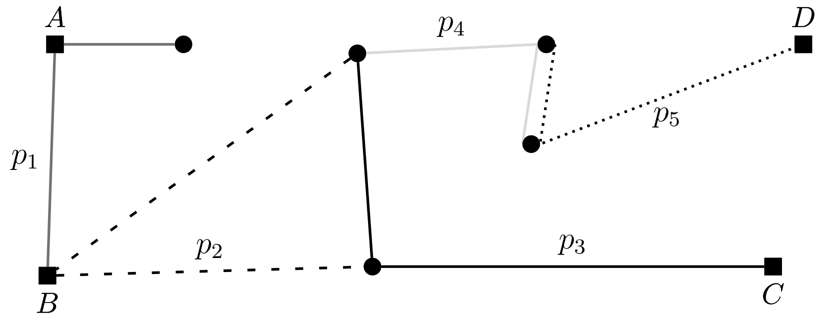

Figure 2: Given is an inclusion wise minimal T-connecting set for $T = \{A, B, C, D\}$. The proof of Lemma 2.1 considers only p_1, p_2, p_3, p_5. A starting order can be $\mathcal{P}'(T) = \{(p_1, p_2, p_3), (p_5)\}$ which is also a final order. We have $T_1 = T_2 = \{A, B\}$, $T_3 = \{A, B, C\}$, $T_4 = \{A, B, C, D\}$. The path p_1 gives rise to case 1., while p_2 comes in case 2. Paths p_2 and p_3 constitute a pair.

p_i, i. e., terminal nodes not contained in T_{i-1}. Then we have one of the following two cases:

1. $r_i \geq 1$; then the maximum number of terminal nodes from the set T_{i-1} contained in path p_i is
   - the minimum of $|T_{i-1}| - 1 = (\sum_{j=1}^{i-1} r_j) - 1$ and $k+1-r_i$, if k is the maximum length of the paths, or
   - the minimum of $|T_{i-1}| - 1 = (\sum_{j=1}^{i-1} r_j) - 1$ and $k - r_i$, if k is the maximum number of terminal nodes.

   In both cases, p_i increases the sum of the path-degrees of all terminal nodes by at most r_i plus the minimum of k and $(\sum_{j=1}^{i-1} r_j) - 1$.
2. $r_i = 0$, i. e., p_i contains a subset of terminal nodes of T_{i-1}. Note that the order of the paths implies that the terminal nodes in p_i are connected by a subset of the paths $p_1, \ldots, p_{i-1}$.
   Due to minimality, there has to exist a path p_h, $h > i$, with $V(p_i) \cap V(p_h) \neq \emptyset$ such that p_h adds $r_h \geq 1$ new terminal nodes and covers no terminal nodes of T_i, i. e., p_i and p_h have a non-terminal node in common. Move path p_h to position $i+1$. Both paths, p_i and p_h, increase the sum of the path-degree of the terminal nodes by at most r_h plus the minimum of $\{|T_i| - 1 = (\sum_{j=1}^{i-1} r_j) - 1,\, k\}$:
   - If k is the maximum path-length, p_i contains at most k terminal nodes since it has a non-terminal node in common with p_h.
   - If k is the maximum number of terminal nodes in a path, the statement above is also true.

The (final) order of the set $\mathcal{P}'(T)$ yields m paths and pairs of paths, respectively, that increase the path-degree on all terminal nodes by at most $\overline{r}_i + \min\{k, (\sum_{j=1}^{i-1} \overline{r}_j) - 1\}$, $\overline{r}_i \geq 1$, $i = 1, \ldots, m \leq n$. Let $1 \leq j \leq m$ be an index such that (i) and (ii) below are satisfied (if $\sum_{i=1}^{m} \overline{r}_i \leq k$, the Lemma holds trivially); clearly (iii) – (v) also hold.

(i) $\sum_{i=1}^{j} \overline{r}_i \geq k+1$,
(ii) $\sum_{i=1}^{j-1} \overline{r}_i \leq k$,

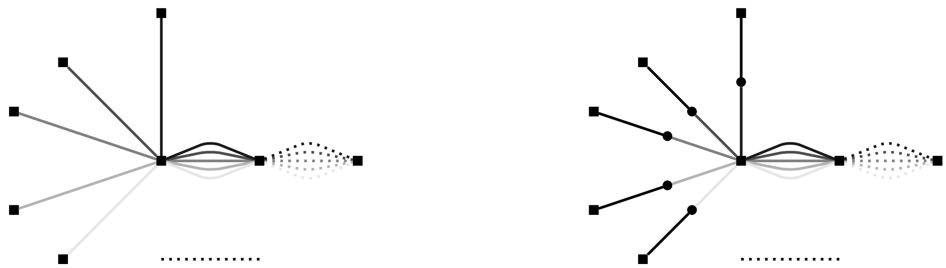

Figure 3: Worst case example for the Degree Lemma 2.1, case (a) (*left*) and case (b) (*right*).

(iii) $\overline{r}_1 + \ldots + \overline{r}_m = |T|$,
(iv) $m \leq |T|$,
(v) $j \leq k + 1$.

We can then bound the sum of the path-degrees on all terminal nodes as follows:

$$\begin{aligned}
\sum_{t \in T} \deg_{\mathcal{P}'}(t) &= \sum_{t \in T} \deg_{\mathcal{P}'(T)}(t) \\
&\leq \overline{r}_1 + (\overline{r}_2 + \min\{k, \overline{r}_1 - 1\}) + \ldots + (\overline{r}_m + \min\{k, (\sum_{i=1}^{m-1} \overline{r}_i) - 1\}) \\
&\leq \overline{r}_1 + \overline{r}_2 + (\overline{r}_1 - 1) + \ldots + \overline{r}_j + (\overline{r}_1 + \ldots \overline{r}_{j-1} - 1) + \overline{r}_{j+1} + k + \ldots \overline{r}_m + k \\
&\overset{\text{(i)-(iii)}}{\leq} |T| + (\overline{r}_1 - 1) + \ldots + (\overline{r}_1 + \ldots + \overline{r}_{j-1} - 1) + (m - j)k \quad (1) \\
&\overset{\text{(ii),(iv)}}{\leq} |T| + k(j-1) - 1 \cdot (j-1) + (|T| - j)k \\
&= |T| + jk - k - j + 1 + |T|k - jk \\
&\overset{\text{if } j \geq 2}{\leq} |T| - 1 + |T|k - k = (|T| - 1)(k + 1).
\end{aligned}$$

For $j = 1$ we have $m - 1 \leq (|T| - \overline{r}_1) \leq (|T| - 2)$ since $\overline{r}_1 = k + 1 \geq 2$ and, therefore,

$$\begin{aligned}
(1) = |T| + (m-1)k &\leq |T| + (|T| - 2)k \\
&\leq |T| + |T|k - k - 1 = (|T| - 1)(k + 1).
\end{aligned}$$

□

We briefly show that this bound is tight for case (a) and (b) of Lemma 2.1. Consider the instance in the left of Figure 3. All nodes are terminal nodes. We have n nodes in the rim and k nodes in 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., the total degree is $n(k+1)$, which gives an average path-degree of $\frac{n(k+1)}{(n+k)}$. This is arbitrarily close to $(k + 1)$ as n goes to infinity. This instance can be slightly modified to get a worst case example for case (b), compare with the right of Figure 3. 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 maximal 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 is analogous to the algorithm of Goemans and Williamson [3] for the Steiner forest problem. Our application to the Steiner connectivity problem is listed in Algorithm 1.

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 (2).

More precisely, let B^i be the set of all connected components in iteration i of Algorithm 1; 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 $B^i_p = \{b \in B^i : p \in \mathcal{P}_{\delta(b)}\}$ 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 $|B^i_p| = 0$ for $B^i_p = \emptyset$. In iteration i we choose, among paths $p \in \mathcal{P}$ with $|B^i_p| > 0$, a path for which the quotient of reduced cost and number of connected components the path cuts in iteration i is minimal, 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 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 1 is based on the consideration of

Algorithm 1: A primal-dual heuristic for the Steiner connectivity problem.

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

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

$B^0 := \{\{t\} \mid t \in T\}\}$, $\mathcal{P}' := \emptyset$, $c^0_p := c_p$ for all $p \in \mathcal{P}$, $i := 0$
$y_W := 0 \, \forall W \subset V, \, W \cap T \neq \emptyset, \, V \setminus W \cap T \neq \emptyset$ //only set when needed
while $|B^i| > 1$ **do**
    $p = \text{argmin}_{q \in \mathcal{P}} \{ \frac{c^i_q}{|B^i_q|} : |B^i_q| > 0 \}$
    $a^i := \frac{c^i_p}{|B^i_p|}$
    $\mathcal{P}' := \mathcal{P}' \cup \{p\}$
    for all $b \in B^i$ **do**
        $y_b := y_b + a^i$
    end
    $B^{i+1} := (B^i \setminus B^i_p) \cup \{b_1 \cup \ldots \cup b_k \cup V(p)\}$ with $\{b_1, \ldots, b_k\} := B^i_p$
    for all $q \in \mathcal{P} \setminus \mathcal{P}'$ **do**
        $c^{i+1}_q := c^i_q - |B^i_q| a^i$
    end
    $i := i + 1$
end
for all $p \in \mathcal{P}'$ **do**
    if $\mathcal{P}' \setminus p$ is T-connecting **then**
        $\mathcal{P}' := \mathcal{P}' \setminus p$ //deleting step
    end
end

the following dual programs:

$$\begin{array}{llll} \min & \displaystyle\sum_{p \in \mathcal{P}} c_p \, x_p & & \\ \text{s.t.} & \displaystyle\sum_{p \in \mathcal{P}_{\delta(W)}} x_p & \geq 1 & \forall W \in \mathcal{W} \\ & x_p & \geq 0 & \forall p \in \mathcal{P} \end{array} \qquad \begin{array}{llll} \max & \displaystyle\sum_{W \in \mathcal{W}} y_W & & \\ \text{s.t.} & \displaystyle\sum_{W \in \mathcal{W} : p \in \mathcal{P}_{\delta(W)}} y_W & \leq c_p & \forall p \in \mathcal{P} \\ & y_W & \geq 0 & \forall W \in \mathcal{W}. \end{array} \tag{2}$$

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. This set of paths has to contain at least one path of each Steiner path cut and is, hence, a T-connecting set.

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 1 gives solutions for the primal and dual programs* (2)*.*

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

$$\sum_{W \in \mathcal{W}: p \in \mathcal{P}_{\delta(W)}} y_W \overset{\text{(i)}}{=} \sum_{i=0}^{r} \sum_{b \in B_p^i} a^i \overset{\text{(ii)}}{=} c_p^0 - c_p^{r+1} \overset{\text{(iii)}}{\leq} c_p. \tag{3}$$

(i) Compare with line 8 in Algorithm 1.

(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 1.

Hence, y is feasible for the dual program in (2). Moreover, we have equality in (3) for $p \in \mathcal{P}'$, i. e.,

$$\sum_{W \in \mathcal{W}: p \in \mathcal{P}_{\delta(W)}} y_W = c_p. \tag{4}$$

□

Proposition 3.2. *Given a Steiner connectivity instance, let $\mathcal{P}_{\text{opt}}$ be the minimum cost T-connecting set and $\mathcal{P}'$ a T-connecting set computed with Algorithm 1. Then*

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

i. e., Algorithm 1 is a $(k+1)$-approximation algorithm with k being the minimum of

(a) the maximal number of edges in a path,

(b) the maximal number of terminal nodes in a path.

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

$$\sum_{p \in \mathcal{P}'} c_p \overset{(4)}{=} \sum_{p \in \mathcal{P}'} \sum_{W \in \mathcal{W}: p \in \mathcal{P}_{\delta(W)}} y_W = \sum_{W \in \mathcal{W}} \sum_{p \in \mathcal{P}_{\delta(W)} \cap \mathcal{P}'} y_W = \sum_{W \in \mathcal{W}} \deg_{\mathcal{P}'}(W) y_W.$$

If we can show the following

$$\sum_{W \in \mathcal{W}} \deg_{\mathcal{P}'}(W) y_W \leq (k+1)\left(1 - \tfrac{1}{|T|}\right) \sum_{W \in \mathcal{W}} y_W, \tag{5}$$

we are done. Note that this is more general than to require $\deg_{\mathcal{P}'}(W) \leq (k+1)(1 - \frac{1}{|T|})$. We show inequality (5) by induction over the iterations of the algorithm. Initially, $y_W = 0$ for all $W \in \mathcal{W}$, so inequality (5) 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

$$a^i \sum_{b \in B^i} \deg_{\mathcal{P}'}(b) \leq (k+1)\big(1 - \tfrac{1}{|B^i|}\big) \sum_{b \in B^i} a^i \overset{|B^i| \leq |T|}{\leq} (k+1)\big(1 - \tfrac{1}{|T|}\big) \sum_{b \in B^i} a^i$$
$$\Leftrightarrow \sum_{b \in B^i} \deg_{\mathcal{P}'}(b) \leq (k+1)(|B^i| - 1) = (k+1)\big(1 - \tfrac{1}{|B^i|}\big)|B^i|.$$

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. The rest follows with the Degree Lemma 2.1 since $\mathcal{P}'$ is minimal. □

References

[1] Ralf Borndörfer and Marika Karbstein, *A Direct Connection Approach to Integrated Line Planning and Passenger Routing*, in ATMOS 2012 - 12th Workshop on Algorithmic Approaches for Transportation Modeling, Optimization, and Systems, Daniel Delling and Leo Liberti, eds., Dagstuhl, Germany, 2012, Schloss Dagstuhl–Leibniz-Zentrum für Informatik, pp. 47–57.

[2] Ralf Borndörfer, Marika Karbstein, and Marc E. Pfetsch, *The Steiner connectivity problem*, Mathematical Programming, (2012). Accepted for publication May 2012.

[3] Michel X. Goemans and David P. Williamson, *A general approximation technique for constrained forest problems*, SIAM Journal on Computing, 24 (1995), pp. 296–317.

[4] Marika Karbstein, *Line Planning and Connectivity*, PhD thesis, TU Berlin, 2013.

[5] Takeshita Kazuhiko, Fujito Toshihiro, and Watanabe Toshimasa, *On primal-dual approximation algorithms for several hypergraph problems*, Joho Shori Gakkai Kenkyu Hokoku, 99 (1999), pp. 13–18.

[6] Karl Nachtigall and Karl Jerosch, *Simultaneous network line planning and traffic assignment*, in ATMOS 2008 - 8th Workshop on Algorithmic Approaches for Transportation Modeling, Optimization, and Systems, Matteo Fischetti and Peter Widmayer, eds., Dagstuhl, Germany, 2008, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, Germany.

[7] Anita Schöbel and Susanne Scholl, *Line planning with minimal traveling time*, in Proceedings of 5th Workshop on Algorithmic Methods and Models for Optimization of Railways, Leo G. Kroon and Rolf H. Möhring, eds., 2006.