New Approximation Algorithms for the Steiner Tree Problems

Marek Karpinski [*] Alexander Zelikovsky [†]

TR-95-036

August 1995

Abstract

The Steiner tree problem asks for the shortest tree connecting a given set of terminal points in a metric space. We design new approximation algorithms for the Steiner tree problems using a novel technique of choosing Steiner points in dependence on the possible deviation from the optimal solutions. We achieve the best up to now approximation ratios of 1.644 in arbitrary metric and 1.267 in rectilinear plane, respectively.

---

[*]Dept. of Computer Science, University of Bonn, 53117 Bonn. Research partially supported by the Leibniz Center for Research in Computer Science, by the DFG Grant KA 67314-1, by the ESPRIT BR Grants 7097 and by ECUS030. Email: marek@cs.uni-bonn.de.

[†]Institute of Mathematics, Akademiei 5, Kishinev, 277028, Moldova. Research partially supported by Volkswagen Stiftung. Parts of this work were done in Max-Planck-Institut für Informatik, Saarbrücken. Email: 17azz@mathem.moldova.su.

1 Introduction

We consider a metric space with a distance function d. For any set of *terminal* points S one can efficiently find MST(S), a minimum spanning tree of S. Let mst(S, d) be the cost of this tree in metric d. A Steiner tree is a spanning tree of a superset of the terminal points (the extra points are called Steiner points). It was already observed by Pierre Fermat that the cost of a Steiner tree of S may be smaller than mst(S, d). The Steiner tree problem asks for the Steiner minimum tree, that is, for the least cost Steiner tree. However, finding such a tree is NP-hard for almost all interesting metrics, like Euclidean, rectilinear, Hamming distance, shortest-path distance in a graph etc. Because these problems have many applications, they were subject of extensive research [12].

In the last two decades many approximation algorithms for finding Steiner minimum trees appeared. The quality of an approximation algorithm is measured by its performance ratio (PR): an upper bound of the ratio between the achieved length and the optimal length.

The Network Steiner tree problem (NSP) asks for the Steiner minimum tree for a vertex subset $S \subset V$ of a graph $G(V, E, d)$ with cost function d on edges E. Let $|V| = v$, $|E| = e$ and $|S| = n$.

In the *rectilinear* metric, the distance between two points is the sum of the differences of their $x-$ and $y-$coordinates. The rectilinear Steiner tree problem (RSP) got recently new importance in the development of techniques for VLSI routing [14].

The most obvious heuristic for the Steiner tree problem approximates a Steiner minimum tree of S with MST(S). While in all metric spaces the performance ratio of this heuristic is at most 2 [17] (it can be implemented for NSP in time $O(e + v \log v)$ [15]), Hwang [10, 11] proved that this heuristic in the rectilinear plane has the performance ratio exactly 1.5 and can be implemented in time $O(n \log n)$.

Consideration of k-restricted Steiner trees gave several better heuristics. For NSP, PR of the em greedy algorithm (GA) (Zelikovsky [18, 20]) is at most $\frac{11}{6} \approx 1.84$ and PR of *Berman-Ramaiyer's heuristic* (BR) [2] is at most $\frac{16}{9} \approx 1.78$. Their runtimes are $O(v^3)$ and $O(\alpha + v^2 n^3)$, respectively (here α means time complexity of finding of all pairs shortest paths). The *relative greedy heuristic* (RGH) (Zelikovsky [21]) with PR converging to $1 + \ln 2 \approx 1.693$ asymptotically beats BR which converges to about 1.734 (Brochers and Du [5]).

In the recent paper Berman *et al* [3] gave a more precise (than in the first papers [19, 2]) analysis of the performance ratio of BR for RSP. They proved that its performance ratio is at most $\frac{61}{48} \approx 1.271$. BR can run in $O(n^{1.5})$ time and its parametrized version (PBR) can run in $O(n \log^2 n)$ time [3, 7].

Here we introduce a novel approach based on the notion of relative gain (see Section 2). Now the choice of Steiner points also depends on the possible deviation from the optimal solution. We add new preprocessing phases to the algorithms mentioned above. Combined algorithms achieve better performance ratios in the same order of the runtime.

The following table contains approximation algorithms known before to be the best in respect to performance ratios and orders of runtime and new performance ratios after preprocessing. By $+\epsilon$ we mean existance of an algorithm for any $\epsilon > 0$.

Problem	Heuristic	Performance Ratio	**New PR**	Run-time	Reference
NSP	MST	2		$O(v^2)$	[17, 15]
	GA	$\frac{11}{6} \approx 1.84$		$O(v^3)$	[18, 20]
	BR	$\frac{16}{9} \approx 1.78$	**253/144≈1.757**	$O(v^5)$	[2]
	RGH	$1 + \ln 2 + \epsilon$ $\approx 1.693 + \epsilon$	**≈1.644+**ϵ	polynomial	[21]
RSP	MST	1.5		$O(n \log n)$	[10, 11]
	BR	$\frac{61}{48} \approx 1.271$	**19/15≈1.267**	$O(n^{1.5})$	[3]
	PBR	$\frac{61}{48} + \epsilon \approx 1.271 + \epsilon$	**≈1.267+**ϵ	$O(n \log^2 n)$	[3]

In the next section we provide a synopsis of k-restricted Steiner trees and our approach. In Sections 3 and 4 we describe our preprocessing of RGH and BR.

2 Gain and Loss of k-Restricted Steiner Trees

2.1 Background

A Steiner tree T of a set of terminals S is *full* if every internal node of T is a Steiner point, i.e., not a terminal. If T is not full, it can be decomposed into full Steiner trees for subsets of terminals that overlap only at leaves. Such subtrees are called *full Steiner components* of T [9]. *k-trees* are full Steiner trees with at most k terminals.

Without loss of generality, we may assume that the metric d on the set of terminals S is the shortest-path distance for the weighted edges D connecting S. This way, $MST(S)$ is the minimum spanning tree of the graph $< S, D >$, we denote this tree by $MST(D)$, and its cost with $mst(D)$ or $mst(S)$. If we increase the set of edges D by some extra edges, say forming a set E, the shortest-path distance may decrease; $MST(D \cup E)$ is the minimum spanning tree for the modified metric. For any graph H, $d(H)$ denotes the sum of costs of all edges of H.

Let $X(T)$ be a Steiner tree obtained from a k-tree T by addition of the minimum forest spanning T with the rest of the terminal set S. The cost of this forest equals to $mst(D \cup E(T))$, where $E(T)$ is the set of zero-cost edges between terminals of T. Define a *gain* of T to be $g(T) = mst(D) - d(X(T)) = mst(D) - mst(D \cup E(T)) - d(T)$. Inductively, the *gain of a set of k-trees* $T_i, i = 1, ..., p$, equals to $mst(D) - d(X(\{T_i, i = 1, ..., p\}))$. Let $R(T)$ denote the set of MST-edges substituted with T in the tree $X(T)$. $R(T)$ consists of the edges of the largest cost on the paths in $MST(D)$ connecting pairs of terminals of T [2]. Denote by $m(T) = mst(D) - mst(D \cup E(T))$ the cost of $R(T)$. Thus, $g(T) = m(T) - d(T)$. Note, that addition of any edges to D may only decrease $m(T)$ and the gain of T [2], therefore,

$$g(\{T_i, i = 1, ..., p\}) \leq \sum_{i=1}^{p} g(T_i). \quad (1)$$

By *contraction* of T we mean addition of $E(T)$ to D. A greedy algorithm (GA) [18] finds a 3-tree with the biggest gain and conracts it while there are 3-trees with a positive gain. All contracted 3-trees and the rest of MST-edges form the output Steiner tree. The k-restricted relative greedy heuristic (k-RGH) [21] runs simalar to GA but maximizing (among all k-trees T) $m(T)/d(T)$ instead of $m(T)-d(T)$. Berman-Ramaiyer [2] suggested a sofisticated generalization of GA for an arbitrary k (k-BR). k-BR proceesses all i-trees, $i = 1, ..., k$, with a positive gain modifying the set D and forming a stack of i-trees chosen. Then it repeatedly pops i-trees from the stack remodifying D and selecting i-trees with the current positive gain. The output tree is $X(T_1, , , T_p)$ for the selected i-trees $T_1, ..., T_p$.

To bound PR of GA, k-RGH and k-BR we need the following constants. Let E_k be an arbitrary set of edges such that in $< S, D \cup E_k >$ the gain of any k-tree becomes nonpositive. We denote by $t_k = t_k(S)$ a supremum of $mst(D \cup E_k)$ over all E_k's.

The output cost of GA [18] ($k = 3$) and k-BR [2] is at most

$$t_2 - \sum_{i=3}^{k} \frac{t_{i-1} - t_i}{i-1} = \frac{t_2}{2} + \sum_{i=3}^{k-1} \frac{t_i}{(i-1)i} + \frac{t_k}{k-1} \tag{2}$$

To bound the values t_k, Berman-Ramayer [2] introduced the following useful definition. A Steiner tree is *k-restricted*, if every its full component is a k-tree. Let $ST_k(S)$ denote a minimal k-restricted Steiner tree and $st_k(S)$ denote its cost. This way, $ST_2(S)$ is the minimum spanning tree $MST(S)$. By (1), the gain of any k-restricted Steiner tree is nonpositive in $< S, D \cup E_k >$, therefore, $st_k(S) \geq t_k(S)$ [2]. These values may not coincide: In the rectilinear plane, for the set $S = \{(\pm 1, 0), (0, \pm 1)\}$, $st_3(S) = 5$ and $t_3(S) = 4.5$.

A *k-Steiner ratio* r_k is the supremum of $st_k(S)/s$ over all instances of the Steiner tree problem, where s denotes the cost of the Steiner minimal tree. r_2 (a usual Steiner ratio) equals 2 and 1.5 for NSP and RSP, respectively [17, 10]. For NSP, some r_k were evaluated in [18, 1, 6] and, finally, Brochers and Du [5] proved that for $k = 2^r + l$,

$$r_k = \sup \frac{st_k}{s} = \frac{(r+1)2^r + l}{r2^r + l}. \tag{3}$$

For the rectilinear metric, $r_k \leq \frac{2k}{2k-1}$ for $r \geq 3$ [2], moreover, for any instance of RSP, $t_2 + t_4 \leq 2.5s$ and $3t_2 + 4t_3 \leq 9s$ [3]. The bounds for t_k and r_k combined with the bound (2) give the preformance guarantee of GA and k-BR mentioned in the previous section. It was proved in [21] that the output cost of k-RGH is at most $(1 + \ln(r_2/r_k))r_k$. Since $\lim_{k\to\infty} r_k = 1$, the limit performance ratio of k-RGH for NSP is at most $1 + \ln 2$. Note that the limit performance ratio of k-BR for NSP derived from (2) and (3) is 1.73...

2.2 A New Approach

The algorithms described above try to maximize the total gain. But every time they accept a k-tree, they also accept all its Steiner points. This may increase the cost of the cheapest solution achievable at the current step. The main idea of our approach is to minimize this possible increase.

Let K be a k-tree and $V(K)$ be its Steiner point set. A forest $K' \subset K$ is called *spanning* if for any $v \in V(K)$, there is a path in K' connecting v with S. The cost of the minimum spanning forest in K is called a *loss* of K and denoted by $l(K)$. The main property of the loss of a k-tree is in the following

Lemma 2.1 *Let P be the set of the Steiner points of an r-tree T. Then $t_k(S \cup P) \leq t_k(S) + l(T)$.*

Proof. Let $< S \cup P, D_P >$ be a complete graph on the set of terminals $S \cup P$ and edges from D_P have costs equal to the shortest-path distances. Let E'_k be an arbitrary set of edges such that $G =< S \cup P, D_P \cup E'_k >$ does not contain k-trees with a positive gain. To prove Lemma it is sufficient to show that $mst(G) \leq t_k(S) + l(T)$.

For every pair of vertices $u, v \in S$, we add an edge $f = (u, v)$ such that $d(f)$ is equal to the largest cost of an edge on the path in $MST(G)$ between u and v. In the graph G' obtained, we can choose a minimum spanning tree M in which any pair $u, v \in S$ is connected by paths containing only terminals of S. It is proved in [2] that the $d(M) = mst(G') = mst(G)$ and for any k-tree K, the cost of $R(K)$ is the same in G and G'.

Consider a subgraph H of G' induced by the vertax set S. Since $MST(H)$ is a subgraph of M, for any k-tree K, $R(K)$ is the same in G' and H. This implies that $g(K)$ is nonpositive in H and $mst(H) \leq t_k(S)$. From the other side, since $S \cup P$ can be spanned with $MST(H)$ and a spanning forest for T, $mst(G) = mst(G') \leq mst(H) + l(T) \leq t_k(S) + l(T) \diamondsuit$

For any $\alpha \geq 0$, the value $g'(\alpha, K) = g(K) - \alpha l(K)$ will be called a *α-relative gain* of K. Further we omit α if $\alpha = 1$. Similarly to the definition of $t_k(S)$, we define $t^k(\alpha) = t^k(\alpha, S)$ to be a supremum of $mst(D \cup E^k)$ over all edge sets E^k's such that addition of E^k to D makes the α-relative gain of any k-tree nonpositive.

Lemma 2.2 $t^k(\alpha, S) \leq (1 + \frac{\alpha}{2}) st_k(S)$

Proof. Let T_i be a full component of an optimal k-restricted Steiner tree T. We transform T_i to the form of a binary tree by replicating certain internal vertices, so that copies of the same vertex are connected with zero-cost edges.

The loss of T_i can be bounded in the following way. For any inner vertex of T_i, choose the cheapest edge among two edges going to its two children. It is easy to see, that the forest F obtained spans all inner vertices of T_i. $d(F)$ is at most half of $d(T_i)$, since F contains exactly half of all edges of T_i and $T_i - F$ contains longer edges. This means, that $l(T_i) \leq 0.5 d(T_i)$.

Let $g(K) \leq \alpha l(K)$ for any k-tree K in $< S, D \cup E^k >$. By (1), $mst(D \cup E^k) - d(T) = g(T) \leq \sum_{i=1}^{p} g(T_i) \leq \sum_{i=1}^{p} \alpha l(T_i) = 0.5 \alpha d(T)$. Therefore, $mst(D \cup E^k) \leq (1 + 0.5\alpha) d(T)$. Since this is true for any E^k, $t^k(\alpha, S) \leq (1 + \frac{\alpha}{2}) d(T) = (1 + \frac{\alpha}{2}) st_k \diamondsuit$.

Theorem 2.2 shows that $\lim_{k \to \infty} t^k(\alpha) = (1 + \frac{\alpha}{2}) s$. The relative gain of any triple is nonpositive, therefore, $t^3 = t_2$. In Sections 5 and 6, we find the tight bounds for t^4 in the case of NSP and RSP, respectively.

Lemma 2.3 *For any instance of NSP, $\frac{t^4}{s} \leq \frac{15}{8}$.*

Lemma 2.4 *For any instance of RSP, $\frac{t^4}{s} \leq \frac{7}{5}$.*

The main idea of preprocessing k-BR and k-RGH is to find some k-trees which are good in respect to the relative gain and to add its Steiner points to initial terminal set before running usual k-BR and k-RGH. Using Lemmas 2.2, 2.3 and 2.4, in Sections 3 and 4, we derive the record performance ratios claimed in Introduction.

Theorem 2.5 *For NSP, there is a polynomial-time approximation algorithm with the performance ratio at most* $1.644... + \epsilon$ *for any* $\epsilon > 0$.

Theorem 2.6 *For NSP, there is an 1.757...-approximation algorithm with a runtime* $O(\alpha + v^2 n^3)$.

Theorem 2.7 *For RSP, for any* $\epsilon > 0$, *there are* $\frac{19}{15}$ *and* $\frac{19}{15} + \epsilon$*-approximation algorithms with runtimes* $O(n^{1.5})$ *and* $O(n \log^2 n)$, *respectively.*

3 Preprocessing the Relative Greedy Heuristic

We suggest the following generalization of k-RGH (k-RGH(α)): While $mst(D) \neq 0$, find and contract a k-tree T minimizing $p(T) = (d(T) + \alpha l(T))/m(T)$. The union of k-trees T obtained forms the output tree.

Theorem 3.1 *k-RGH(α) finds a tree T such that* $d(T) + \alpha l(T) \leq (1 + \ln \frac{mst(S)}{t^k(\alpha,S)}) t^k(\alpha, S)$.

Proof. Let $T_1, ..., T_a$ be the k-trees chosen by k-RGH(α) including 2-terminal trees (edges). Let M_j denote $mst(D \cup E(T_1) \cup ... \cup E(T_j))$, $j = 0, ..., a$. Let $p(T_1)D$ be the set of edges D with the cost $p(T_1)$ times the cost of edges of D. Since $p(e) = 1$ for any MST-edge, $p(T_i) \leq 1$ and $MST(D \cup p(T_1)D) = MST(p(T_1)D)$. By the choice of T_1, $< S, p(T_1)D >$ does not contain k-trees with the positive α-relative gain. Therefore, $p_1 mst(D) = mst(D \cup p_1 D) \leq t^k(\alpha)$ and

$$\frac{d(T_1) + \alpha l(T_1)}{m(T_1)} \leq \frac{t^k(\alpha)}{M_0}$$

Similarly, after contracting of T_1 and choosing T_2, we obtain

$$\frac{d(T_2) + \alpha l(T_2)}{m(T_2)} \leq \frac{t^k(\alpha)}{M_1}$$

Note, that $M_i = M_{i-1} - m(T_i)$. Inductively we obtain for each $i \geq 1$, $(d(T_i) + \alpha l(T_i))/(M_{i-1} - M_i) \leq t^k(\alpha)/M_{i-1}$, or equivalently $M_i \leq M_{i-1}(1 - (d(T_i) + \alpha l(T_i))/t^k(\alpha))$. Unraveling these inequalities,

$$M_r \leq M_0 \prod_{i=1}^{r} (1 - \frac{d(T_i) + \alpha l(T_i)}{t^k(\alpha)}).$$

Taking natural logarithm on both sides and using the fact that $\ln(1 + x) \leq x$, we obtain

$$\frac{\sum_{i=1}^{r} (d(T_i) + \alpha l(T_i))}{t^k(\alpha)} \leq \ln \frac{M_0}{M_r}.$$

Since $M_{|S|} = 0$, we can choose r such that $M_r > t^k(\alpha, S) \geq M_{r+1}$. We split $d(T_{r+1}) + \alpha l(T_{r+1})$ proportionally by the position of $t^k(\alpha)$ in the interval $[M_{r+1}, M_r]$. We combine the first portion with M_{r+1} to bring this cost up to exactly $t^k(\alpha)$, and combine the second portion with $d(T_r) + \alpha l(T_r)$. We then split $M_r - M_{r+1}$ into the same proportions, and subtract the second portion from M_r so that the last inequality above still holds when we "pretend" that $t^k(\alpha) = M_{r+1}$. We now finish the proof with the sequence of inequalities

$$\frac{\sum_{i=1}^{a}(d(T_i) + \alpha l(T_i))}{t^k(\alpha)} \leq \frac{M_{r+1}}{t^k(\alpha)} + \frac{\sum_{i=1}^{r+1}(d(T_i) + \alpha l(T_i))}{t^k(\alpha)} \leq 1 + \ln \frac{M_0}{M_{r+1}} = 1 + \ln \frac{mst(S)}{t^k(\alpha, S)} \diamondsuit$$

Now we preprocess k-RGH (k-RGH(0)) with l-RGH(α) in the following way. We run l-RGH(α) obtaining a Steiner tree T and add all Steiner points of T to the initial terminal set S. Then we apply k-RGH to the modified terminal set.

Proof of Theorem 2.5. Our goal is to obtain the limit performance ratio of k-RGH after preprocessing with l-RGH(α) while $l, k \to \infty$. Denote by S_l the modified terminal set after preprocessing and by s_l the cost of the optimal Steiner tree for S_l. Note that $mst(S_l) = d(T)$.

By Lemma 2.2 and Theorem 3.1, while $l \to \infty$, the bound for $(d(T) + \alpha l(T))/s$ converges to

$$B = (1 + \frac{\alpha}{2})(1 + \ln \frac{2}{1 + \frac{\alpha}{2}}). \tag{4}$$

By Theorem 3.1 and Lemma 2.1, the cost of the output of k-RGH applied to S_l is at most

$$(1 + \ln \frac{mst(S_l)}{t_k(S_l)})t_k(S_l) \leq (1 + \ln \frac{d(T)}{t_k(S) + l(T)})(t_k(S) + l(T)). \tag{5}$$

Since $\lim_{k\to\infty} t_k(S) = s$, (4) and (5) imply that the limit output cost is at most

$$(1 + \ln \frac{d(T)}{s + \frac{1}{\alpha}(Bs - d(T))})(s + \frac{1}{\alpha}(Bs - d(T))). \tag{6}$$

As a function of $d(T)$, (6) has one maximum for $d(T)$ such that

$$\frac{(\alpha + B)s - d(T)}{d(T)} = \ln \frac{\alpha d(T)}{(\alpha + B)s - d(T)}.$$

Denote by $f(\alpha)$ the solution of the equation $x = \ln(\alpha/x)$. Then we obtain the following upper bound for the limit output cost

$$f(\alpha)(1 + B/\alpha)$$

The last function has a minimum for $\alpha \approx 0.5$ which is about 1.644... Thus, k-RGH preprocessed with l-RGH(0.5) has a limit performance ratio at most 1.644... while $l, k \to \infty$. $\diamondsuit$

4 Preprocessing Berman-Ramaiyer's Algorithm

An *r-restricted Berman-Ramaiyer's preprocessing* (r-BRP) differs from the usual r-BR only in the gain function substituted with the relative gain function.

Lemma 4.1 *Let $T(r)$ be an output tree of r-BRP. Then $g'(T(r)) \geq \sum_{i=3}^{r} \frac{t^{i-1}-t^i}{i-1}$.*

Proof (Sketch). For any r-tree K, denote $d'(K) = d(K) + l(K)$. For any r-restricted Steiner tree T, $d'(T) = \sum_{i \in A} d'(K_i)$, where $K_i, i \in A$ are the full components of T. Since $d'(T) = d(T)$ for any 2-restricted Steiner tree T, an optimal in respect to d' Steiner tree has a cost at most the minimum spanning tree cost.

r-BRP coincides with r-BR applied to the modified cost function d' instead of d. Berman and Ramaiyer [2] proved that the output tree of the usual r-BR has a gain at least $\sum_{i=3}^{r} \frac{t_{i-1}-t_i}{i-1}$ (compare with (2)). This proof does not use any properties of the cost function d on k-trees except the properties above. Thus, we may conclude that the same fact is true for r-BR applied to the cost function d'.

Since the gain function in respect to d' equals to the relative gain function in respect to d, the relative gain of the the output tree of r-BRP is at least $\sum_{i=3}^{r} \frac{t'_{i-1}-t'_i}{i-1}$, where t'_i means the value of t_i in respect to the cost function d'. Lemma follows from the fact that the value t'_i coincides with the value t^i in respect to the function d for any $i = 1, ..., r$. $\diamondsuit$

Let S_r be the union of the terminal set S with the set of all Steiner points of $T(r)$. Denote by G, L and $G' = G - L$ the total gain, loss and relative gain of $T(r)$, respectively. Then $t_2(S_r) = t_2(S) - G$ and $t_i(S_r) \leq t_i(S) + L$ by Lemma 2.1.

Let bound the cost of the output of k-BR applied to S_r. By (2), it is at most

$$\sum_{i=3}^{k-1} \frac{t_2(S_r) + t_i(S_r)}{(i-1)i} + \frac{t_2(S_r) + t_k(S_r)}{k-1} \leq \sum_{i=3}^{k-1} \frac{t_2(S) - G' + t_i(S)}{(i-1)i} + \frac{t_2(S) - G' + t_k(S)}{k-1}$$

$$= \frac{t_2 - G'}{2} - \sum_{i=3}^{k} \frac{t_{i-1} - t_i}{i-1} \qquad (7)$$

Lemma 4.1 and (7) imply

Theorem 4.2 *The cost of the output Steiner tree of k-BR preprocessed with r-BRP is at most*

$$\frac{t_2}{2} - \sum_{i=3}^{k} \frac{t_{i-1} - t_i}{i-1} - \frac{1}{2} \sum_{i=3}^{r} \frac{t^{i-1} - t^i}{i-1} \qquad (8)$$

Proof of Theorems 2.6 and 2.7. Note that r-BRP has the same order of runtime as r-BR since r-trees with a positive relative gain should have a positive gain and a loss of an r-tree can be found very fast using a greedy algorithm. By Lemma 2.3 and Theorem 4.2, 4-BR preprocessed with 4-BRP satisfies Theorem 2.6.

In the rectilinear metric, the output length of 4-BR preprocessed with 4-BRP can be bounded using Lemma 2.4 and inequalities (8), $3t_2 + 4t_3 \le 9s$ and $2t_2 + 2t_4 \le 5s$. Indeed, this length is at most

$$t_2 - \frac{t_2 - t_3}{2} - \frac{t_2 - t_4}{3} - \frac{1}{2}\frac{t_2 - t^4}{3} = \frac{t_2}{3} + \frac{t_3}{6} + \frac{t_4}{3} + \frac{t^4}{6} \le$$
$$\frac{3t_2 + 4t_3}{24} + \frac{t_2 + t_4}{3} + \frac{t^4}{24} \le \frac{3}{8}s + \frac{5}{6}s + \frac{7}{120}s = \frac{19}{15}s\diamondsuit$$

5 The value of t^4 for NSP

Proof of Lemma 2.3. Further assume that some terminals are connected with short edges such that $g(K) \le l(K)$ for any 4-tree K. We may prove Lemma for each full Steiner component separately. We transform such a component to the form of the complete binary tree by replicating certain vertices, so that copies of the same vertex are connected with zero-cost edges. Note that all terminals are leaves of this tree.

Let k be the depth of this tree. We label its vertices with words from $B^* = \{\alpha \in B^* : |\alpha| \le k\}$, where $B = \{0, 1\}$. Let ρ be the root and α have children $\alpha 0$, $\alpha 1$. The set of terminals with the common anchestor α is denoted by α also.

Some more denotations: Let $s = s(\rho)$ denote the cost of the Steiner minimal tree, $t = t(\rho)$ be the cost of MST for the whole terminal set, $s_i(\alpha) = \sum_{|\beta|=i, b\in B} d(\alpha\beta, \alpha\beta b)$, $H = H(\rho) = s_0(\rho) + s_1(\rho)$, $P(\alpha)$ denote the cost of the cheapest path from α to S.

An *average path cost* is defined to be

$$\bar{P} = \bar{P}(\rho) = \frac{\sum_{i=1}^{k-1} 2^{k-i} s_i(\rho)}{2^k} = \sum_{i=1}^{k-1} 2^{-i} s_i(\rho)$$

This cost has the following two obvious properties:

$$\bar{P}(\alpha) \ge P(\alpha) \tag{9}$$

$$2\bar{P}(\alpha) = s_0(\alpha) + \bar{P}(\alpha 0) + \bar{P}(\alpha 1). \tag{10}$$

Since $\bar{P} \ge \frac{H}{4}$, the following inequality is slightly stronger than Lemma.

$$t \le 2s - 2\bar{P} - \frac{s - H}{8} \tag{11}$$

We will prove (11) by induction on k. Indeed, for $k \le 2$, (11) is trivially true. Let (11) be true for all trees of depth at most k. We will prove it for a tree of depth $k + 1$ (Fig. 1).

Further assume that $s_1(0) \ge s_1(1)$.

Now we partition $s(\rho)$ into five subtrees:

$$s(\rho) = \sum_{\alpha\in A} s(\alpha) + D,$$

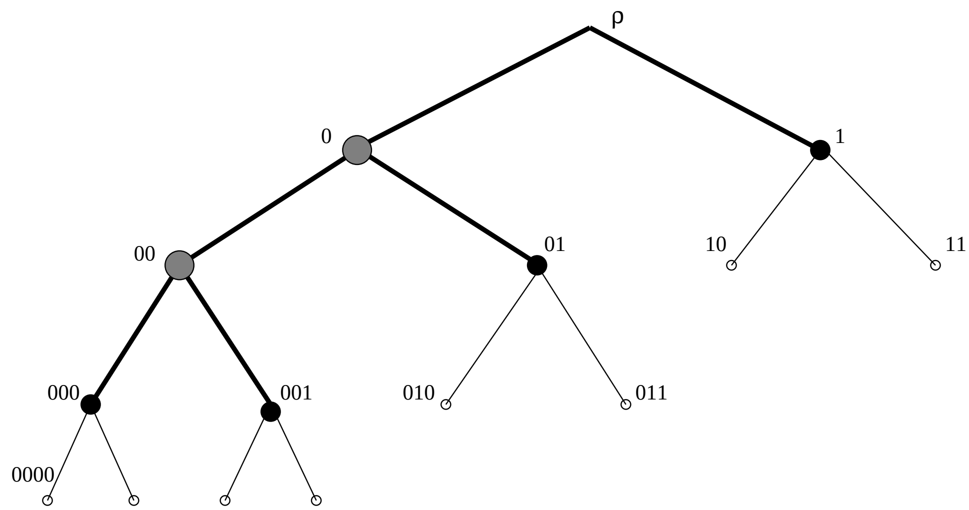

Figure 1: A full component

where $\alpha \in A = \{000, 001, 01, 1\}$ and $D = s_0(\rho) + s_0(0) + s_0(00)$ (thick lines on Fig. 1).

These five parts correspond to some spanning tree:

$$t(\rho) \leq \sum_{\alpha \in A} t(\alpha) + t', \tag{12}$$

where t' is the cost of three cheapest edges connecting four MST for the sets $\alpha \in A$. By induction, inequality (11) holds for every $\alpha \in A$:

$$t(\alpha) \leq 2s(\alpha) - 2\bar{P}(\alpha) - \frac{s(\alpha) - H(\alpha)}{8} \tag{13}$$

Substituting (13) into (12) we obtain

$$t(\rho) \leq 2(s - D) - 2\sum_{\alpha \in A} \bar{P}(\alpha) - \sum_{\alpha \in A} \frac{s(\alpha) - H(\alpha)}{8} + t'$$

and, therefore,

$$t(\rho) - (2s - 2\bar{P} - \frac{s - H}{8}) \leq t' + 2\bar{P} + \frac{s - H}{8} - 2D - 2\sum_{\alpha \in A} \bar{P}(\alpha) - \sum_{\alpha \in A} \frac{s(\alpha) - H(\alpha)}{8}.$$

To prove (11) it is sufficient to show that the RHS of the last inequality is nonpositive, which is equivalent to the following inequality

$$\frac{1}{8}\left(s - H - \sum_{\alpha \in A}(s(\alpha) - H(\alpha))\right) \leq 2D + 2\sum_{\alpha \in A} \bar{P}(\alpha) - (t' + 2\bar{P}) \tag{14}$$

Claim 5.1 *The RHS of (14) is at least* $\bar{P}(0) - d(0,00)$.

Proof. Consider an arbitrary 4-tree q with Steiner points 0 and 00 and four terminals achievable from 000, 001, 01 and 1, respectively. Note, that $t' \leq t(q)$, where $t(q) = d(q) + g(q)$ is the cost of three corresponding longest edges on paths connecting treminals of q. Let terminals of q be the nearest to the corresponding vertices of A. Since $g(q) \leq l(q) \leq d(0,00) + P(00)$, we obtain

$$t' \leq D + \sum_{\alpha \in A} P(\alpha) + d(0,00) + P(00)$$

Now Claim can be proved straitforward using the properties (9) and (10) of the average path cost:

$$2D + 2\sum_{\alpha\in A} \bar{P}(\alpha) - (t' + 2\bar{P}) \geq$$

$$2D + 2\sum_{\alpha\in A} \bar{P}(\alpha) - (D + \sum_{\alpha\in A} P(\alpha) + d(0,00) + P(00) + s_0(\rho) + \bar{P}(0) + \bar{P}(1)) \geq$$

$$s_0(0) + s_0(00) + \bar{P}(000) + \bar{P}(001) + \bar{P}(01) - P(00) - \bar{P}(0) - d(0,00) \geq \bar{P}(0) - d(0,00) \quad \diamondsuit$$

The LHS of (14) equals to $\frac{1}{8}(D + \sum_{\alpha\in A} H(\alpha) - H) = \frac{1}{8}(s_1(1) + s_0(01) + s_1(01) + s_0(00) + s_1(00) + s_2(00))$. By Claim and our assumption of $s_0(00) + s_0(01) = s_1(0) \geq s_1(1)$, (14) follows from the following inequality

$$\frac{1}{8}(2s_0(01) + s_1(01) + 2s_0(00) + s_1(00) + s_2(00)) \leq \bar{P}(0) - d(0,00) \tag{15}$$

Similarly, the corresponding partition of the Steiner minimal tree induced by the 4-tree with Steiner points 0 and 01 implies that it is sufficient to prove

$$\frac{1}{8}(2s_0(00) + s_1(00) + 2s_0(01) + s_1(01) + s_2(01)) \leq \bar{P}(0) - d(0,01) \tag{16}$$

Thus, to prove (11) we may show that one of the inequalities (15) or (16) is true. This follows from the fact that their sum is true. Indeed, summing (15) and (16) we obtain

$$\frac{1}{8}(4s_0(00) + 2s_1(00) + s_2(00) + 4s_0(01) + 2s_1(01) + s_2(01)) \leq 2\bar{P}(0) - s_0(0) = \bar{P}(00) + \bar{P}(01),$$

which trivially follows from the definition of the average path cost. $\diamondsuit$

6 The value of t^4 for RSP

Hwang [10] proved that there is a Steiner minimum tree where every full component has one of the shapes shown in Fig. 2. It was suggested in [3] some partition of a full component into so called Steiner segments. Below we breifly describe this useful technique.

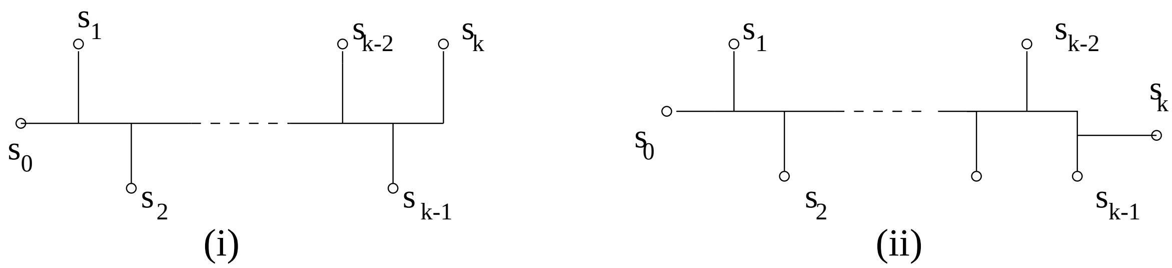

Figure 2: Two types of a full component

Let $a_1, \ldots, a_k$ and $b_0 = 0, b_1, \ldots, b_k$ be the lengths of horizontal and vertical lines of a full Steiner component F with terminals $s_0, \ldots, s_k$. The horizontal lines form its *spine*. Moreover, in case (i) $b_k < b_{k-2}$ holds. In case (ii) assume that $b_k = 0$. Consider the sequences $b_0, b_1, b_3, \ldots, b_{2i+1}, \ldots$ and $b_0, b_2, \ldots, b_{2i}, \ldots$. Let

$$b_{h(0)} = b_0, b_{h(1)}, \ldots, b_{h(p+1)} = b_k \tag{17}$$

be the sequence of local minima of these sequences, i.e. $b_{h(j)-2} \geq b_{h(j)} < b_{h(j)+2}$. If $h(p) = k-1$, we exclude the member $b_{h(p)}$ from (17). For the case of $h(j+1) = h(j)+1$, $(j = 1, \ldots, p-1)$, we exclude arbitrarily either $b_{h(j+1)}$ or $b_{h(j)}$. So, we get $h(j+1) - h(j) \geq 3$. The elements of the refined sequence (17) are called *hooks*. Further we assume that a full Steiner tree nontrivially contains at least 4 terminals ($k \geq 4$). A *Steiner segment* K is a part of a full Steiner component bounded by two sequential hook terminals. So two neighbouring Steiner segments have a common hook. K contains the two furthest terminals below and above the spine called *top* and *bottom*, respectively.

Now we are ready to start the following

Proof of Lemma 2.4. Further assume that some terminals are connected with short edges such that $g(K) \leq l(K)$ for any 4-tree K. It is sufficient to prove Lemma for a full Steiner component F with a terminal set Set. Let $F = \cup_{i=0}^{k} K_i$ be a partition of F into Steiner segments. Then $d(F) = \sum_{i=0}^{k} d(K_i) - \sum_{i=1}^{k-1} h_i$, where h_i are hooks. Consider some Steiner segment $K = K_i$ of F with terminal set $S = S_i$, hooks $hl = h_i$ and $hr = h_{i+1}$ and the length $s = d(K)$. Similarly to Section 5, denote the MST-length for a terminal set X by $t(X)$. We intend to prove that

$$t(S) - s \leq \frac{2}{5}s - \frac{7}{10}(hl + hr) \tag{18}$$

This inequality yields Lemma, since then

$$t(Set) \leq \sum_{i=0}^{k} t(S_i) \leq \frac{7}{5}\sum_{i=0}^{k} d(K_i) - \frac{7}{10}\sum_{i=0}^{k-1}(h_i + h_{i+1}) \leq$$

$$\frac{7}{5}(\sum_{i=0}^{k} d(K_i) - \sum_{i=1}^{k-1} h_i) = \frac{7}{5}d(F)$$

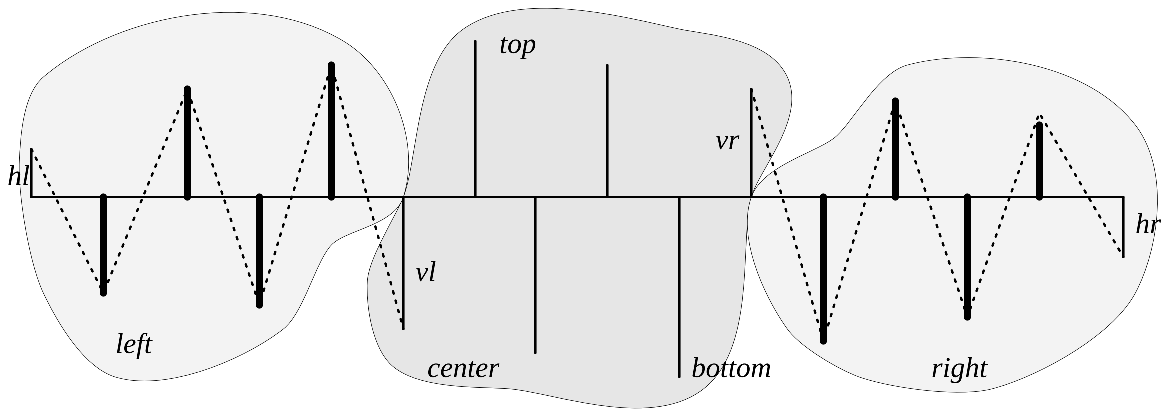

Figure 3: The partition of the Steiner segment

Let *top* of K be to the left of its *bottom*. We partition S into three parts $S = L \cup C \cup R$, where L is the set of terminals from the left hook till the first before top, C contains all terminals from the the first before top till the next after bottom and R contains ones from the next after bottom till the right hook. Similarly, we partition F into three corresponding parts

$$s = left + center + right,$$

where $center$ contains all edges spanning C, and $left$ and $right$ consists of the rest of the Steiner segment to the left and right of $center$ (Fig. 3). Denote by vl and vr the lengths of two vertical lines which bound $center$ from the left and the right. Note that K should contain $center$, but $left$ and $right$ might be empty.

We have two cases depending on the size of $center$.

Case 1. Let *bottom* be the next to *top* (Fig. 4). For this case we need the following useful

Lemma 6.1 *[3] There are two trees (Fig. 4(i)) Top (dashed lines) and Bot (dotteded lines) spanning terminals of K with a total length*

$$d(Top) + d(Bot) = 3s - 2(hl + hr) - Rest;$$

$Rest$ sums the lengths of the thin drawn Steiner tree lines.

Lemma 6.1 says that $t \leq \frac{3}{2}s - \frac{Rest}{2} - (hl + hr)$. It is easy to see that (18) holds if $Rest$ is big enough, i.e. $Rest \geq \frac{s}{5} - \frac{3}{5}(hl + hr)$. So further assume that

$$Rest \leq \frac{s}{5} - \frac{3}{5}(hl + hr). \tag{19}$$

We may span R and L with the alternative chains (Fig. 3), therefore,

$$t(L) + t(R) \leq left + right + Rest - x, \tag{20}$$

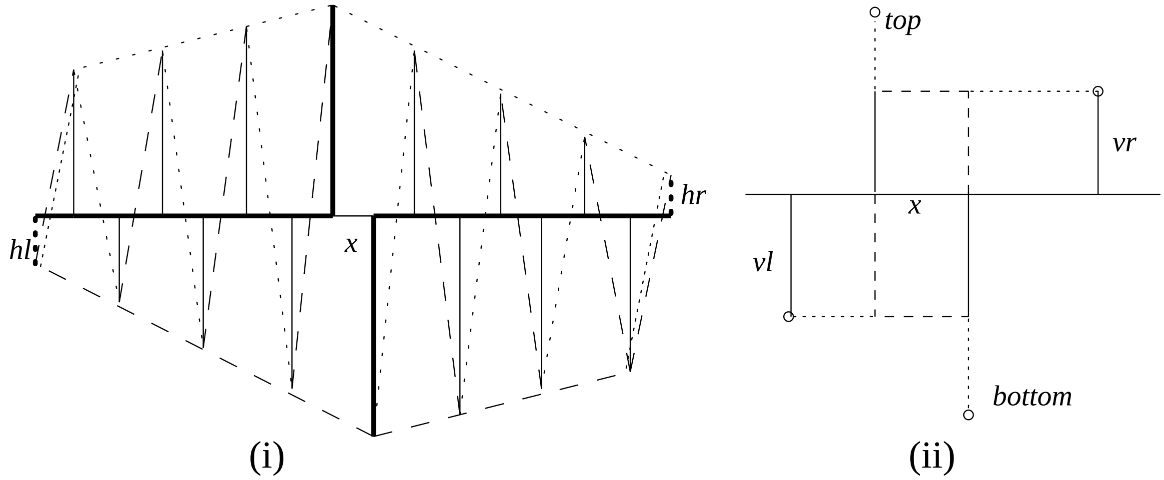

Figure 4: *top* besides *bottom*: the whole segment (i) and its *center* (ii)

where x is the horisontal edge length of $Rest$.

Let q be the quadruple with terminals from C (Fig. 4 (ii)). Lemma assumes that $g(q) = t(C) - center$ is at most $l(q)$. But the loss of q is at most x plus the length of the shortest among four dotted lines (we may shift the central edge up or down till dashed lines). Therefore,

$$t(C) - center \leq l(q) \leq x + \frac{center - (2vl + 2vr + x)}{4} \leq x + \frac{s - Rest - (hl + hr)}{4} \qquad (21)$$

Thus, we can prove (18) using (19), (20), (21):

$$\begin{aligned} t(S) - s &= (t(C) - center) + (t(L) - left + t(R) - right) \\ &\leq x + \frac{s - Rest - (hl + hr)}{4} + Rest - x \leq \frac{s}{4} + \frac{3}{4}Rest - \frac{hl + hr}{4} \\ &\leq \frac{s}{4} + \frac{3}{4}(\frac{s}{5} - 3\frac{hl + hr}{5}) - \frac{hl + hr}{4} = \frac{2}{5}s - \frac{7}{10}(hl + hr) \end{aligned}$$

Case 2. Let two terminals lie between *top* and *bottom*. Now *center* contains two quadruples $q1$ and $q2$ with central edges $x1$ and $x2$ (Fig. 5). We construct 5 spanning trees for the set C. Three trees contain some connection of the quadruple $q1$ and pairs of edges spanning the last two terminals: thick dotted, dashed, and solid lines, respectively. Lemma assumes that the connection of the quadruple $q1$ cannot be longer the length of $q1$ (Steiner edges in the dark region) plus the loss of $q1$. Denote by *light* the length of Steiner edges out of the dark region. Then

$$T1 - center \leq d(q1) + l(q1) + light + a + h3 - center = l(q1) + a + h3 \leq x1 + c + a + h3$$

$$T2 - center \leq l(q1) + h2 + d \leq h1 + b + h2 + d$$

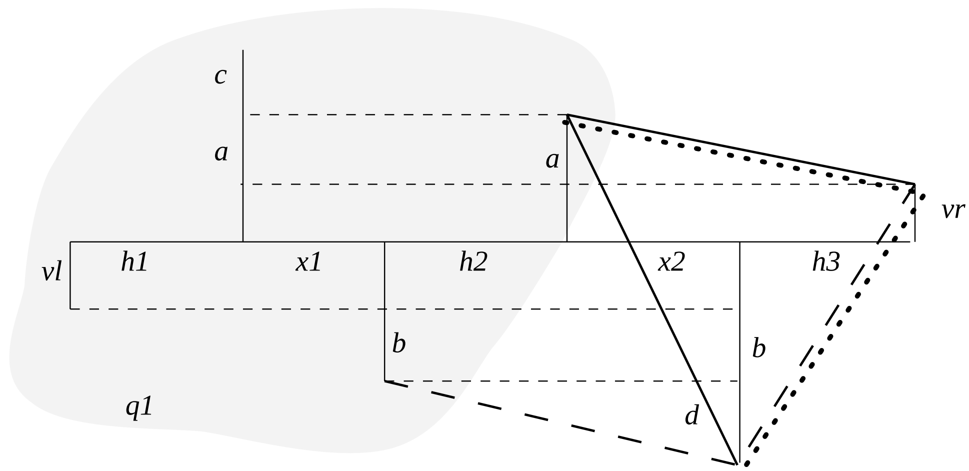

Figure 5: 2 terminals between *top* and *bottom*

$$T3 - center \leq l(q1) + 2a + x2 \leq x1 + b + 2a + x2$$

The last pair of trees is symmetric to $T1$ and $T2$

$$T4 - center \leq l(q2) + b + h1 \leq x2 + d + b + h1$$

$$T5 - center \leq l(q2) + h2 + c \leq h3 + a + h2 + c$$

Summing all inequalities we obtain

$$5t(C) - 5center \leq 2center - 6(vl + vr) \tag{22}$$

If there are more terminals between *top* and *bottom* then *center* contains several quadruples q_i. Three necessary spanning trees contain connections of odd qudruples and two contain connections of even quadruples. Similarly, we obtain (22) using the Lemma assumption that such connections are no longer than $d(q_i) + l(q_i)$.

To prove (18), we will show that

$$5(t(L) + t(R)) - 5(left + right) \leq 2(left + right) - 4(hl + hr) + 6(vl + vr),$$

which means for the right side of the Steiner segment

$$5t(R) - 5right \leq 2right - 4hr + 6vr \tag{23}$$

If vr is the right hook ($vr = hr$), then (23) is trivial, since $t(R) = right = 0$.

If the hook is the next after vr (Fig. 6(i)), then we use the solid line five times and two times replace the edge of $T1$ and $T2$ (the thick dashed line) with the dotted line. In the latter case we replace vr and hr with f, the horizontal edge length. Thus, we obtain $5t(R) - 5right \leq 5vr + 2f - 2hr \leq 2right - 4hr + 6vr$.

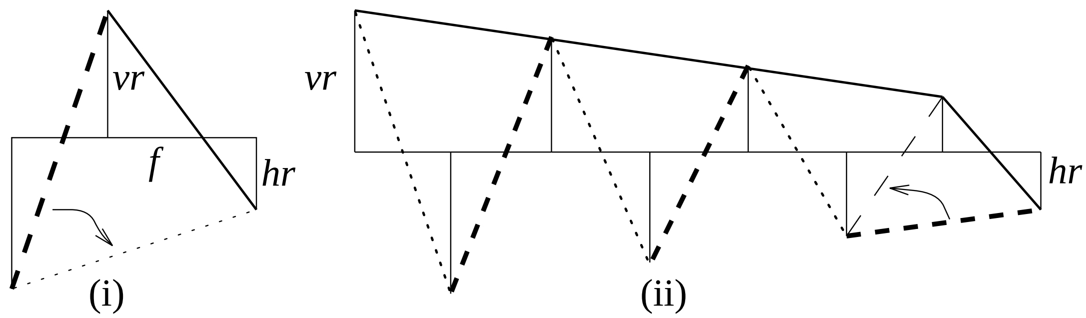

Figure 6: The short (i) and the long (ii) *right*

For a nontrivial R we use the following 5 trees (Fig. 6(ii)) which contain:
(1) thick solid and dotted lines. It doubles vr and Steiner tree lines crossed by its dotted lines.
(2-3) thick solid and dashed lines or the thin dashed line if the hook is above the spine (2 times). It doubles the Steiner tree lines crossed by its edges and saves the hook hr.
(4-5) the alternative chain (Fig. 3) (2 times). It doubles all vertical lines except vr and hr.

Thus, these trees double $right - hr$ at most two times, vr only once, and save hr two times. $\diamondsuit$

7 Conclusion and Open Problems

The main open question remaining for the Network Steiner Tree Problem is to compute the exact value of a constant c which separates polynomial approximability from non-approximability (NP-hardness) of this problem. Such a constant c must exist since NSP is $MAX\ SNP$-complete [4]. We prove that c lies somewhere below 1.644... for that problem. Note that we do not know at the moment whether RSP is also $MAX\ SNP$-complete, and therefore it could have a polynomial time approximation scheme. At the end a word about achieved heuristics: Our paper shows for the first time that we are able to solve with at most 26.7% error any practical instance of RSP of size, say, up to 10^5 in 1 h, whereas all other known algorithms of the same quality are able to solve RSP only for about 30 points in 24 hours on a SUN3 workstation (see [16]).

References

[1] P. Berman, V. Ramaiyer. *An approximation algorithm for the Steiner tree problem* . TR #CS-91-05, The Pennsylvania State University, 1991.

[2] P. Berman, V. Ramaiyer. *Improved approximations for the Steiner tree problem.* J. of Algorithms 17: 381–408, 1994.

[3] P. Berman, U. Fößmeier, M. Karpinski, M. Kaufmann, A. Zelikovsky. *Approaching the 5/4–Approximations for Rectilinear Steiner Trees.* LNCS 855, 60–71, 1994.

[4] M. Bern and P. Plassmann. *The Steiner problems with edge lengths 1 and 2.* Inform. Process. Lett. 32: 171–176, 1989.

[5] A. Borchers and D. Du. *The k-Steiner Ratio in Graphs* To appear in STOC'1995.

[6] D. Du, Y. Zhang, Q. Feng. *On better heuristic for Euclidean Steiner minimum trees.* 32nd IEEE Symp. on Found. of Comp. Science, 431–439, 1991.

[7] U. Fößmeier, M. Kaufmann, A. Zelikovsky. *Faster Approximation Algorithms for the Rectilinear Steiner Tree Problem.* LNCS 762, 533-542, 1993.

[8] M. R. Garey, D. S. Johnson. *The Rectilinear Steiner Problem is NP-Complete.* SIAM J. Appl. Math., 32, 826-834, 1977.

[9] E. N. Gilbert, H. O. Pollak. *Steiner Minimal Trees.* SIAM Appl. Math., 16, 1-29, 1968.

[10] F. K. Hwang. *On Steiner Minimal Trees with Rectilinear Distance.* SIAM J. Appl. Math., 30, 104-114, 1976.

[11] F. K. Hwang. *An $o(n \log n)$ algorithm for rectilinear minimum spanning trees.* J. ACM 26, 177–182, 1979.

[12] F. K. Hwang, D. Richards, P. Winter. *The Steiner Tree Problem.* Annals of Disc. Math. 53, 1992.

[13] M. Karpinski, A. Zelikovsky. *1.757 and 1.267-Approximation Algorithms for the Network and Rectilinear Steiner Tree Problems*, Technical Report TR–95–010, International Computer Science Institute, Berkeley, 1995.

[14] B.Korte, H.J.Prömel, A.Steger. *Steiner Trees in VLSI-Layouts.* In Korte et al.: Paths, Flows and VLSI-Layout, Springer, 1990.

[15] K.Mehlhorn. *A faster approximation algorithm for the Steiner problem in graphs.* Inf. Process. Lett. 27: 125–128, 1988.

[16] J.S.Salowe and D.M.Warme. *An exact rectilinear Steiner tree algorithm.* Proc. of the Int. Conf. on Comp. Design, 472–475, 1993.

[17] H. Takahashi, A. Matsuyama. *An approximate solution for the Steiner problem in graphs.* Math. Japonica, 24: 573–577, 1980.

[18] A. Z. Zelikovsky. *An 11/6-approximation Algorithm for the network Steiner Problem* Algorithmica, 9: 463–470, 1993.

[19] A. Z. Zelikovsky. *An 11/8-approximation Algorithm for the Steiner Problem on Networks with Rectilinear Distance.* In Coll. Math. Soc. J. Bolyai 60: 733-745, 1992.

[20] A. Z. Zelikovsky. *A faster approximation algorithm for the Steiner Tree Problem in Graphs.* Inf. Process. Lett. 46: 79-83, 1993.

[21] A. Z. Zelikovsky. *Better approximation bounds for the network and Euclidean Steiner tree problems* 1995.