Approximation schemes for node-weighted geometric Steiner tree problems

Journal Article

Author(s):
Remy, Jan; Steger, Angelika

Publication date:
2009-09

Permanent link:
https://doi.org/10.3929/ethz-b-000157324

Originally published in:
Algorithmica 55(1), https://doi.org/10.1007/s00453-007-9114-6

Algorithmica (2009) 55: 240–267
DOI 10.1007/s00453-007-9114-6

Approximation Schemes for Node-Weighted Geometric Steiner Tree Problems

Jan Remy · Angelika Steger

Received: 4 September 2007 / Accepted: 1 October 2007 / Published online: 31 October 2007

Abstract In this paper we introduce a new technique for approximation schemes for geometrical optimization problems. As an example problem, we consider the following variant of the geometric Steiner tree problem. Every point u which is not included in the tree costs a penalty of $\pi(u)$ units. Furthermore, every Steiner point that we use costs c_S units. The goal is to minimize the total length of the tree plus the penalties. Our technique yields a polynomial time approximation scheme for the problem, if the points lie in the plane.

Keywords Computational geometry · Steiner tree problem · Approximation schemes

1 Introduction

1.1 Approximation Schemes for Geometric Problems

Let $\mathcal{P}$ denote a set of points in $\mathbb{R}^d$ and let $d_q(u,v)$ denote the distance between $u \in \mathcal{P}$ and $v \in \mathcal{P}$ in the $\mathcal{L}_q$-metric. In the sequel we consider graphs on $\mathcal{P}$, like for instance spanning trees and salesman tours. The length $\ell_q(e)$ of an edge $e = \{u,v\}$ in such a graph is the distance $d_q(u,v)$ between its endpoints. A natural question is to ask for the shortest spanning tree or the shortest salesman tour on $\mathcal{P}$ in the $\mathcal{L}_q$-metric. Here, 'shortest' refers to the total length of the edges appearing in the corresponding structure. Such optimization problems are not known to be in $\mathcal{NP}$, since they

A preliminary version of this paper appeared in the *Proceedings of the 8th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems*, 2005, 221–232.

J. Remy (✉) · A. Steger
Institut für Theoretische Informatik, ETH Zürich, Switzerland
e-mail: jremy@inf.ethz.ch

A. Steger
e-mail: steger@inf.ethz.ch

involve the computation of square roots. However, it is possible to compute a shortest spanning tree for $\mathcal{P}$ in polynomial time using standard algorithms. In contrast, no such algorithm is known for the geometric traveling salesman problem. In fact, this problem was shown to be strongly $\mathcal{NP}$-hard by Garey, Graham and Johnson in the late 70ies [11]. The same authors proved in [12] that also the geometric Steiner tree problem, which is to find the shortest tree connecting $\mathcal{P}$, is strongly $\mathcal{NP}$-hard. Note that a Steiner tree may include additional points, the so-called Steiner points, and can thus be shorter than the minimum spanning tree. As these problems are strongly $\mathcal{NP}$-hard, a polynomial time approximation scheme (PTAS) is the best we can hope for, unless $\mathcal{P} = \mathcal{NP}$.

There is a vast literature on approximation algorithms for geometrical problems. The progress until 1996 is discussed in the survey of Bern and Eppstein in [7]. An important breakthrough was made in 1996 when Arora [2, 3] and Mitchell [18, 19] independently introduced (different) polynomial time approximation schemes for the geometric traveling salesman problem, and the geometric Steiner tree problem, and other $\mathcal{NP}$-hard problems in the plane. Later Arora extended his method to arbitrary fixed dimension [3]. In case of the traveling salesman problem this is nearly best possible, as Trevisan [27] showed that there exists no PTAS in dimension $\mathcal{O}(\log n)$, unless $\mathcal{P} = \mathcal{NP}$.

Arora's technique applies to a wide class of problems for which the so-called Patching Lemma holds, which essentially states that one can reduce the number of crossings over a given line segment without much increase in the cost of the solution. However, there are quite a number of geometric problems, where it seems unclear whether such a result is true. This is, for example, the case for the k-MEDIAN problem, MINIMUM WEIGHT TRIANGULATION, MINIMUM WEIGHT STEINER TRIANGULATION or BOUNDED-DEGREE MINIMUM SPANNING TREE. Nevertheless, for some of these problems polynomial time approximation schemes were recently obtained by different methods. For example, we have a PTAS for k-MEDIAN [6, 17] and a quasi-polynomial time approximation scheme (QPTAS) for both the BOUNDED-DEGREE MINIMUM SPANNING TREE problem [5] and MINIMUM WEIGHT TRIANGULATION problem [22]. Recall that it is well-known, that the existence of a QPTAS implies that the problem is not $\mathcal{APX}$-hard, provided SAT $\notin$ DTIME$[n^{\mathrm{polylog}(n)}]$.

1.2 Our Contribution

The main aim of our paper is to provide a variant of Arora's approximation scheme that avoids the use of the Patching lemma. From a technical viewpoint (see Sect. 3 for details) this is achieved by characterizing partial solutions by their structure within a subrectangle instead of by their intersection with the boundary of the rectangle. We explain our technique by developing an approximation scheme for the NODE-WEIGHTED GEOMETRIC STEINER TREE problem, a generalization of the geometric Steiner tree problem. In the plane, our method yields a PTAS for the problem. For (fixed) dimensions $d \geq 3$, we are still able to derive a QPTAS.

1.3 Outline

The remainder of the paper is organized as follows. In Sect. 2 we formally introduce the problem NODE-WEIGHTED GEOMETRIC STEINER TREE (NWGST). In Sect. 3

we outline the basic principles of our algorithmic strategy. In Sect. 4 and Sect. 5 we provide fundamental definitions and properties of (restriced) optimal solutions of NWGST. In Sect. 6 we then present and analyse the polynomial time approximation scheme. Finally, in Sect. 7 we extend the algorithm to higher dimensions and present further generalizations.

2 The Node-Weighted Geometric Steiner Tree Problem

2.1 Problem Statement

An instance of the node-weighted geometric Steiner tree problem in $\mathbb{R}^d$, or NWGST(d) for short, consists of a set of points $\mathcal{P}$ in $\mathbb{R}^d$, a function $\pi : \mathcal{P} \to \mathbb{Q}_+$ and a $c_S \in \mathbb{Q}_+$. A solution $\mathcal{ST}$ is a (geometric) spanning tree on $V(\mathcal{ST}) \cup S(\mathcal{ST})$, where $V(\mathcal{ST}) \subseteq \mathcal{P}$ and $S(\mathcal{ST}) \subset \mathbb{R}^d$. The points in $S(\mathcal{ST})$ are called *Steiner points*. Let $E(\mathcal{ST})$ denote the set of line segments or edges used by $\mathcal{ST}$. We minimize

$$\mathrm{val}\,(\mathcal{ST}) = \sum_{\{u,v\}\in E(\mathcal{ST})} d_q\,(u,v) + \sum_{u\in\mathcal{P}-V(\mathcal{ST})} \pi(u) + |S(\mathcal{ST})| c_S. \tag{1}$$

The objective function can be intuitively understood as follows. We pay for the total length of the tree plus a penalty of c_S units for every Steiner point we use. Furthermore, we are charged $\pi(u)$ units for every point $u \in \mathcal{P}$ which is not included in the tree. In network design, for instance, additional switches could reduce the cable length of the computer network but they are expensive to install.

Recall that it is easy to compute a spanning tree of minimum length. The main difficulty in the node weighted Steiner tree problem is therefore the choice of appropriate sets $V(\mathcal{ST})$ and $S(\mathcal{ST})$. Throughout the paper $\mathcal{ST}^*$ denotes the optimal solution for the input set $\mathcal{P}$. For the sake of brevity we will also use the following notations in this paper. For a solution $\mathcal{ST}$, $\ell_q(\mathcal{ST})$ denotes the total length of the tree $\mathcal{ST}$. Furthermore, $\pi(\mathcal{ST})$ and $c_S(\mathcal{ST})$ denote the total amount of penalties we pay for unconnected points and the use of Steiner points, respectively. That is, (1) rewrites to $\mathrm{val}(\mathcal{ST}) = \ell_q(\mathcal{ST}) + \pi(\mathcal{ST}) + c_S(\mathcal{ST})$.

NWGST(d) covers several well-known problems as special cases. If we choose $c_S = 0$ and $\pi(u) = \infty$ then we obtain the *geometric Steiner tree problem* [9, 14]. Furthermore, by choosing just $c_S = 0$, we have the prize-collecting variant of the geometric Steiner tree problem. If both $c_S = \infty$ and $\pi(u) = \infty$ then NWGST(d) is equivalent to MINIMUM SPANNING TREE. In this paper we will prove

Theorem 1 *For fixed $\varepsilon > 0$, there is a polynomial time algorithm which computes a $(1+\varepsilon)$-approximation to* NWGST(2).

This means that NWGST(2) and thus also the PRIZE-COLLECTING GEOMETRIC STEINER TREE admit a polynomial time approximation scheme (PTAS). In Sect. 7 we show

Theorem 2 *For fixed $\varepsilon > 0$ and fixed $d \geq 3$, there is a quasi-polynomial time algorithm which computes a $(1+\varepsilon)$-approximation to* NWGST(d).

In other words, the problem admits a quasi-polynomial time approximation scheme in arbitrary fixed dimensions. In this section, we also discuss how far our method extends to variants of the problem.

2.2 Related Work

The input of the *Steiner tree problem in networks* is a weighted graph $\mathcal{G} = (V, E)$ and a set $K \subseteq V$. The goal is to find a tree of minimum weight in $\mathcal{G}$ which includes all vertices in K. This problem is known to be $\mathcal{APX}$-complete [8] and it does therefore not admit a PTAS unless $\mathcal{P} = \mathcal{NP}$. However, there are many constant-factor approximation algorithms for this problem. The currently best algorithm is by Robins and Zelikovsky [23]. It yields a $(1 + \frac{\ln 3}{2})$-approximation. There is also a node-weighted version of this problem [14, 24] which is similar to our model. It was shown by Klein and Ravi [16] that there is an approximation algorithm with performance ratio $2 \ln |K|$. Furthermore, they proved that it is not possible to achieve a ratio better than logarithmic, unless SAT $\in$ DTIME[$n^{\text{polylog}(n)}$]. A similar setup was considered by Moss and Rabani [20]. In essence they combined the unweighted Steiner tree problem with packing problems. A 2-approximation algorithm for the prize-collecting variant of the Steiner tree problem in networks is due to Goemans and Williamson [13].

As already mentioned the geometric Steiner tree problem admits a PTAS [3, 19] and is strongly $\mathcal{NP}$-hard [12]. Thus, from complexity theoretic point of view there is not much room for improvements. Arora claimed in [4] that there is a QPTAS for the prize collecting variant. Talwar [26] shows that Arora's method extends to metrics that satisfy certain properties but the complexity of his approximation scheme is quasi-polynomial. The Steiner tree problem is extensively discussed in the textbooks of Hwang, Richards and Winter [14] and Prömel and Steger [21].

3 Outline of the Algorithm

In this section we outline our algorithmic strategy for NWGST(2). First, we briefly review the main ideas of Arora's approximation scheme for the geometric Steiner tree problem. Arora subdivides the smallest rectangle enclosing all points recursively by using a quadtree such that the rectangles of the leaves of the quadtree contain at most one point. The structure of a Steiner tree $\mathcal{ST}$ inside a rectangle of this quadtree can be described by specifying (i) the locations where edges of $\mathcal{ST}$ cross the boundary of the rectangle and (ii) how these locations are connected inside the rectangle. Both parameters define the configuration of this rectangle. Given a rectangle and a configuration, Arora's algorithm optimizes locally as follows. It computes the best solution for this configuration by enumerating all configurations of the children of the rectangle and combining their optimum solutions. As the optimum solutions for rectangles containing at most one point are easy to compute, the optimum solutions of all rectangles can be computed bottom-up by dynamic programming. In other words, a look-up table is maintained which contains for each configuration-rectangle pair the corresponding optimum. The complexity of this dynamic program depends on the size of this table, i.e., the size of the quadtree and the number of different configurations per rectangle. We shall see later that the quadtree has logarithmic height and contains therefore polynomially many rectangles. In order to reduce the number of configurations per rectangle, one has to do some rounding. This is achieved as follows.

Firstly, edges are only allowed to cross the boundary of a rectangle at one out of $\mathcal{O}(\log n)$ prespecified locations, the so-called portals. Secondly, the tree may cross the boundary of a rectangle at most constantly many times. In this way one can show that there exist only polynomially many different configurations for each rectangle and the dynamic program therefore terminates in polynomial time. On the other hand, by reducing the number of configurations, the dynamic program optimizes only over a subclass of all trees. One therefore also has to show that this subclass contains a tree such that its length differs only slightly from that of an optimum tree. By using Arora's Patching Lemma and by introducing randomness into the quadtree one can show that this is indeed the case.

Unfortunately, our problem NWGST(2) seems not to fit in this framework, as the fact that additional Steiner points add additional costs to the tree makes it hard to imagine that an optimum tree can always be changed in such a way that it crosses the boundary of all rectangles only at constantly many positions. We therefore use a completely different approach to define configurations. Instead of specifying the locations where the tree edges cross the boundary of a rectangle, we aim at specifying the structure of the tree within a rectangle and at specifying at which locations it should be connected to points outside of the rectangle. Basically, this is achieved by subdividing the rectangle into $\mathcal{O}(\log n)$ many cells. A configuration specifies which cells contain an endpoint of an edge crossing the boundary and specifies to which component inside of the rectangle those points belong. In this way we restrict the number of different locations of end points of edges crossing the boundary of a rectangle, but not the number of such edges.

4 Preliminaries

In the sequel we mainly consider an integral variant of NWGST(2) which we denote by NWLST(2). The difference to NWGST(2) is that we require that all input points have odd integral coordinates and that the side length of the bounding box is $L = \mathcal{O}(n^2)$. That is, we have $\mathcal{P} \subset \{1, 3, 5, \ldots, L-1\}^2$. Similarly, the Steiner points contained in the solution must also have odd integral coordinates. This is similar to the notion of well-rounded instances in [3].

Lemma 3 *If there is a PTAS for* NWLST(2) *then there is also a PTAS for* NWGST(2).

For the proof of Lemma 3 we need an approximation algorithm for NWGST(2) with constant performance ratio. This motivates the following statement.

Lemma 4 *There is an algorithm which computes in polynomial time a* 3*-approximation to* $\mathcal{ST}^*$.

Proof By using the algorithm of Goemans and Williamson [13], we compute a 2-approximation to the optimal prize-collecting spanning tree on $\mathcal{P}$. Let A denote the total cost, i.e., length plus penalties, of this approximation. We claim that

$$A \leq 3 \cdot \operatorname{val}\left(\mathcal{ST}^*\right).$$

Let $\mathcal{ST}'$ denote the minimum spanning tree on $V(\mathcal{ST}^*)$ and note that $c_S(\mathcal{ST}') = 0$. It was shown by Du and Hwang [10] that for every set of points $\mathcal{P}$ the length of the MST is at most $3/2$ the length of the optimal Steiner tree. Thus, $\mathcal{ST}'$ satisfies

$$\begin{aligned}\operatorname{val}\left(\mathcal{ST}'\right) = \ell_q\left(\mathcal{ST}'\right) + \pi\left(\mathcal{ST}'\right) &\leq \frac{3}{2} \cdot \ell_q\left(\mathcal{ST}^*\right) + \pi\left(\mathcal{ST}^*\right) + c_S\left(\mathcal{ST}^*\right) \\ &\leq \frac{3}{2} \cdot \operatorname{val}\left(\mathcal{ST}^*\right),\end{aligned}$$

since $V(\mathcal{ST}') = V(\mathcal{ST}^*)$ and thus $\pi(\mathcal{ST}') = \pi(\mathcal{ST}^*)$. Clearly, the cost of the optimal prize-collecting spanning tree $\mathcal{PCT}^*$ on $\mathcal{P}$ is at most val$(\mathcal{ST}')$. Since the algorithm described above computes a 2-approximation to the optimum prize-collecting MST, we obtain

$$A \leq 2 \cdot \operatorname{val}\left(\mathcal{PCT}^*\right) \leq 2 \cdot \operatorname{val}\left(\mathcal{ST}'\right) \leq 3 \cdot \operatorname{val}\left(\mathcal{ST}^*\right). \qquad \square$$

Proof of Lemma 3 Note that we can improve every solution to NWGST(2) by deleting Steiner points that are either leaves or have just two neighbors in the tree. Hence, every Steiner tree which is optimal in this sense contains at most n Steiner points and thus at most $2n$ points. Let A denote the value of the solution obtained by the approximation algorithm of Lemma 4. Furthermore, let M denote the side length of the bounding box of $\mathcal{P}$. We scale c_S, the penalties and all coordinates by

$$\rho = \frac{24n}{\varepsilon A}$$

and move every point (including the Steiner points) to the nearest points with odd integral coordinates. This means that all points lay on a grid of granularity 2 which we call the 2-grid. We denote this rounded point set by $\mathcal{P}'$. If more points are moved to a grid point then we treat them as single point. The penalty associated with this point is the sum of the penalties of the points moved to this coordinate. The side length of the bounding box is now $L = M\rho$. By moving the points we may have increased the length of the optimum solution. However, we move at most $2n$ points by at most 4 units. Since we have at most $2n - 1$ edges, we obtain

$$\operatorname{val}\left(\mathcal{ST}^{*\prime}\right) \leq \rho \cdot \operatorname{val}\left(\mathcal{ST}^*\right) + 8n,$$

where $\mathcal{ST}^{*\prime}$ is the optimal Steiner tree on $\mathcal{P}'$. Assume for the moment, we have a solution $\mathcal{ST}'$ for $\mathcal{P}'$ such that

$$\operatorname{val}\left(\mathcal{ST}'\right) \leq (1+\varepsilon)\operatorname{val}\left(\mathcal{ST}^{*\prime}\right) \leq (1+\varepsilon)\left(\rho \cdot \operatorname{val}\left(\mathcal{ST}^*\right) + 8n\right). \tag{2}$$

Transforming $\mathcal{ST}'$ into a solution $\mathcal{ST}$ for $\mathcal{P}$ can be achieved by moving the points back to their original position and by scaling by $1/\rho$. If $\mathcal{P}'$ contains n' points, the length of each of at most $2n'-1$ edges in $\mathcal{ST}'$ increases by at most 4 units. Moreover, if points $q_1, \dots, q_k \in \mathcal{P}$ were treated as a single point, we keep q_1 in the tree and connect each of $q_2, \dots, q_k$ to q_1 by an edge of length at most 4. Thus we have

$$\mathrm{val}\,(\mathcal{ST}) \le \frac{1}{\rho}\left(\mathrm{val}\left(\mathcal{ST}'\right) + 8(n'-1) + 4(n-n')\right) < \frac{1}{\rho}\left(\mathrm{val}\left(\mathcal{ST}'\right) + 8n\right).$$

That is, we have

$$\begin{aligned}\mathrm{val}\,(\mathcal{ST}) &\le \frac{1}{\rho}\left(\mathrm{val}\left(\mathcal{ST}'\right) + 8n\right) \le \frac{1}{\rho}\left((1+\varepsilon)\cdot\left(\rho\cdot\mathrm{val}\left(\mathcal{ST}^*\right) + 8n\right) + 8n\right)\\ &= (1+\varepsilon)\cdot\mathrm{val}\left(\mathcal{ST}^*\right) + \frac{1}{\rho}8n(2+\varepsilon) = (1+\varepsilon)\cdot\mathrm{val}\left(\mathcal{ST}^*\right) + \frac{1}{3}\varepsilon A(2+\varepsilon)\\ &\le \left(1+3\varepsilon+\varepsilon^2\right)\cdot\mathrm{val}\left(\mathcal{ST}^*\right).\end{aligned}$$

If we want a $(1+\varepsilon')$ approximation for $\mathcal{P}$, we have to choose ε appropriately.

It remains to be shown that we can find a solution $\mathcal{ST}'$ satisfying (2). If $A \ge M/n$ then $L \le 24n^2/\varepsilon$ and we are done. Since $L = \mathcal{O}(n^2)$ for fixed ε, we can simply use the PTAS for NWLST(2).

Thus assume that $A < M/n$. Consider a grid of granularity $g = 24n^2/\varepsilon$ such that the grid lines have even integral coordinates. We call this grid the g-grid. Clearly, there are at most $g^2/4 = \mathcal{O}(n^4)$ such grids. We claim that there exists at least one g-grid such that the (scaled) optimum solution is completely contained within a box of this grid. Thus, we can enumerate all g-grids, treat every box in such a g-grid as a separate instance, apply the PTAS for NWLST(2) to it, and finally return the best solution obtained during this process.

It remains to show that such a grid exists. To see this, we choose the grid uniformly at random. Every edge e of the scaled optimum solution crosses lines of the 2-grid. One can easily check that the number of lines crossed is a lower bound for the weight of the scaled optimum (note that the lines have even coordinates and that $\ell_q(e) \ge \ell_\infty(e) \ge \ell_1(e)/2$). Thus, the optimum solution, scaled by ρ, intersects at most ρA lines of the 2-grid. On the other hand, the probability that a fixed line of the 2-grid is also used in the g-grid is at most $2/g$. Thus, the probability that the optimal Steiner tree crosses a line of g-grid is at most

$$\rho A \frac{2}{g} = \frac{2}{n}.$$

Therefore, there is at least one g-grid such that the optimum solution is completely contained within a box of this grid. □

Next, we adapt the concept of *shifted quadtrees* or *shifted dissections* [3, 4] to our purposes. We require that L is a power of 2. If this is not the case, we simply enlarge the bounding box appropriately. We choose two integers a and b with $a, b \in$

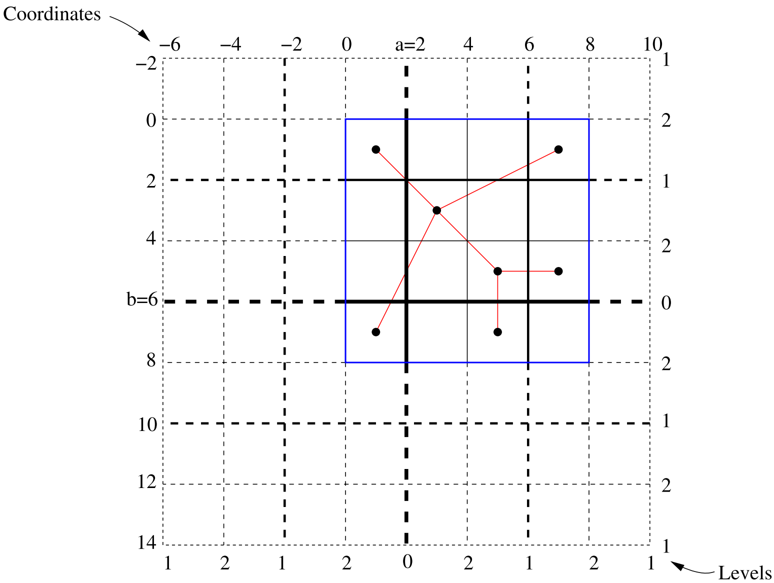


Fig. 1 The shifted quadtree. The *dashed lines* indicate the enlarged rectangles at level 0 and the *dissection lines* within

$\{0, 2, \dots, L-2\}$. The vertical line with x-coordinate a and the horizontal line with y-coordinate b split the bounding box into four smaller rectangles. We enlarge those rectangles such that their side length is L. By $\mathfrak{S}_0$ we denote the area covered by the four enlarged rectangles. Note that $\mathfrak{S}_0$ is a square of side length $L_0 = 2L$. We now subdivide $\mathfrak{S}_0$ using an ordinary quad tree until we obtain squares of size 2×2. We denote the quadtree generated in this fashion by $QT_{a,b}$ to emphasize that its structure depends on the choice of a and b.

Throughout we denote vertical dissection lines with *even* x-coordinates by V and horizontal ones with *even* y-coordinates by H. A vertical line V with x-coordinate x has *level* l *with respect to* a if there is an *odd* (potentially negative) integer i, such that $x = a + i \cdot \frac{L}{2^l}$. Throughout, $\text{lev}(V, a) = l$ denotes the smallest l for which such an i exists. Similarly, we define the level of horizontal lines and $\text{lev}(H, b)$. The *level* of a square $\mathfrak{S}$ in the quadtree $QT_{a,b}$ is defined as follows: The squares in the very first subdivision have level 0 and a square $\mathfrak{S}$ at level l is subdivided by a horizontal line H and a vertical line V with $\text{lev}(H, b) = \text{lev}(V, a) = l + 1$ into four squares at level $l + 1$. The level of a square $\mathfrak{S}$ is denoted by $\text{lev}(\mathfrak{S})$. Let us shortly summarize salient properties of shifted quadtrees.

Observation 5 *For a shifted quadtree $QT_{a,b}$ the following is true*:

(1) *The subdivision uses only horizontal and vertical lines that have* even *coordinates.*
(2) *The depth of $QT_{a,b}$ is* $\log L =: t$.
(3) *The side length of a square $\mathfrak{S}$ at level l is $S = L/2^l$.*
(4) *The number of vertical* (*horizontal*) *lines in $QT_{a,b}$ at level l is at most 2^l.*

Note that t is integral by choice of L. This property will turn out to be useful in Sect. 5.1. Figure 1 illustrates the definitions we made above as well as Observation 5.

5 (s,t)-Maps and Standardized Solutions

5.1 (s,t)-Maps

From now on, we only consider the problem NWLST(2). As already mentioned, we need some subdivision of a square $\mathfrak{S} \in QT_{a,b}$ into cells. A natural idea would be to use a regular $(m \times m)$-grid. Indeed, such grids already appear in quasi-polynomial approximation schemes of [6, 26]. Unfortunately, it will later turn out that we would have to choose

$$m = \Omega(st) = \Omega(\log n)$$

to obtain a $(1 + 1/s)$-approximation, where $s = \mathcal{O}(1/\varepsilon)$. Therefore, this grid has $\Omega(\log^2 n)$ many cells. This is too much, since we later store some bits per square and since we require that we have polynomial many states per square.

This problem is solved by using so-called (s,t)-maps which have only $\mathcal{O}(\log n)$ cells. The intuition behind (s,t)-maps is very simple. In the PTAS, special care has to be taken for edges that cross boundaries of squares. Due to complexity, we do not explicitly specify the endpoints of such edges. Instead, we will argue that it suffices to describe them as 'edges' between cells. This has the effect that our PTAS can only reconstruct that the endpoints are somewhere within the cells. However, the length of the edge varies by at most twice the side length ($\mathcal{L}_1$-metric) of the cells. The definition of (s,t)-maps comes directly from the following simple observation. We can estimate the endpoints of long edges more roughly than those of short edges, since we have only to assure that the absolute error is at most $1/s$ of the edge's optimal length. Edges that reach deep into $\mathfrak{S}$ are long and thus we can make the cells inside $\mathfrak{S}$ larger than those which are close to the boundary. The definition of an (s,t)-*map* follows exactly this idea. We will ensure that the side length of a cell $\mathfrak{C}$ is at most

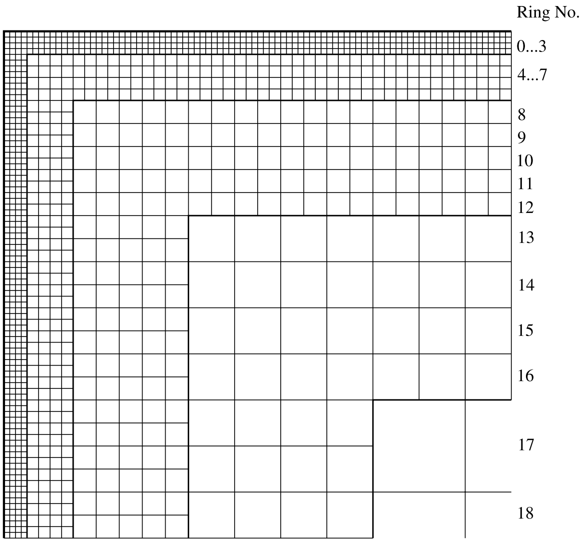
$$\max\left\{\frac{1}{st}\langle\text{side length of } \mathfrak{S}\rangle, \frac{1}{s}\langle\text{distance between } \mathfrak{C} \text{ and boundary of } \mathfrak{S}\rangle\right\}$$

and that the map has $\mathcal{O}(s^2 t)$ cells. In essence, an (s,t)-map can be regarded as a $(st \times st)$-grid with cell sizes growing to the interior.

Now we define (s,t)-maps more formally. For arbitrary $s \in \mathbb{N}$, we define a function $\beta : \mathbb{N}_0 \to \mathbb{N}_0$ as

$$\beta(j) = \begin{cases} \left\lfloor \frac{j}{4s} \right\rfloor, & j < 8s, \\ 2, & 8s \le j \le 9s - 1, \\ \left\lfloor \frac{j-s}{4s} \right\rfloor, & j \ge 9s. \end{cases}$$

Fig. 2 The upper-left part of an (s,t)-map with $s=1$. The cell size doubles for each group of rings. Each group has $4s=4$ rings except the 3rd group which has $5s=5$ rings



Let $m=m(s)=8s2^{\tau}$. Here τ is the smallest integer such that $2^{\tau}\geq t$, where t is the depth of the shifted quadtree. Note that this implies $\tau<1+\log t$. Therefore, we have $8st\leq m<16st$.

Let S be the side length of $\mathfrak{S}$. We subdivide $\mathfrak{S}$ in *rings*, as illustrated in Fig. 2 for $s=1$. The jth ring has width $2^{\beta(j)}S/m$ and consists of congruent axis parallel squares of the same size. We call this subdivision the (s,t)-*map* and write $\mathcal{M}[\mathfrak{S}]$. We have groups of $4s$ rings of same size. The only exception is the third group which contains $5s$ rings. Since we later require that the cells partition the square, we assume that the upper and left boundary always belong to the corresponding neighboring cell if there is any.

Let γ count the number of groups. Since the width of a ring in the ith group is $2^{i-1}S/m$, γ must satisfy $(5s-4s)2^2\frac{S}{m}+4s\sum_{i=1}^{\gamma}\frac{2^{i-1}S}{m}\overset{!}{=}\frac{S}{2}$, and thus $\gamma=\log(m/8s)=\tau\leq 1+\log t$. Note that γ is integral by definition of τ. In the remainder of this section we state some properties of (s,t)-maps. In some sense, (s,t)-maps and grids share salient properties. However, the number of cells in a $m\times m$ grid is $m^2=\Theta(s^2t^2)$ while an (s,t)-map has much fewer cells.

Lemma 6 *$\mathcal{M}[\mathfrak{S}]$ contains $\mathcal{O}(s^2t)$ cells.*

Proof In each of the $4s$ outermost rings, we have less than $4m$ cells per ring. Since the size of the cells doubles from group to group, we have at most

$$4ms+\sum_{i=1}^{\gamma}\frac{16ms}{2^{i-1}}\leq 36ms=\mathcal{O}\left(s^2t\right)$$

cells in $\mathcal{M}[\mathfrak{S}]$, where the first term counts the number of the s many additional rings in the third group. $\square$

Lemma 7 *Let $\mathfrak{C} \in \mathcal{M}[\mathfrak{S}]$ have side length $C \geq 2S/m$ and let $p \in \mathcal{P}$ be a point which is contained in $\mathfrak{C}$. Then the distance of p to the boundary of $\mathfrak{S}$ in any $\mathcal{L}_q$-metric is at least $2sC$.*

Proof Assume that $C = 2^{k-1}S/m$ for $k \geq 2$, i.e. $\mathfrak{C}$ is in the kth group. Then the minimum distance of p to the boundary of $\mathfrak{S}$ is simply the orthogonal distance to the nearest boundary line of $\mathfrak{S}$. This is true for any $\mathcal{L}_q$-metric. In order to obtain a lower bound to this distance, it suffices to add up the widths of the rings between $\mathfrak{C}$ and its nearest boundary of $\mathfrak{S}$. This sum is at least

$$\sum_{i=1}^{k-1} 4s\frac{2^{i-1}S}{m} = 2^{k+1}s\frac{S}{m} - 4s\frac{S}{m} \geq 2^k s\frac{S}{m} = 2sC$$

for all $k \geq 2$. □

Let $\mathfrak{S}$ be a square at level l and let $\mathcal{M}[\mathfrak{S}]$ be an (s,t)-map on $\mathfrak{S}$. Furthermore, let $\mathfrak{S}'$ be a child of $\mathfrak{S}$ in $QT_{a,b}$ and let $\mathfrak{C}'$ be a cell of $\mathcal{M}[\mathfrak{S}']$. A cell $\mathfrak{C}$ in $\mathcal{M}[\mathfrak{S}]$ is a *parent of* $\mathfrak{C}'$ if $\mathfrak{C}' \cap \mathfrak{C} \neq \emptyset$, i.e. if the cells overlap. We write child$[\mathfrak{C}]$ for the set of children of $\mathfrak{C}$. The next lemma states that the parent-child relation is unambiguous and that all the children of a cell are contained in the same rectangle of $QT_{a,b}$.

Lemma 8 *Every cell has a unique parent. Moreover, for each cell $\mathfrak{C} \in \mathcal{M}[\mathfrak{S}]$, we have* child$[\mathfrak{C}] \subseteq \mathcal{M}[\mathfrak{S}']$ *for some child $\mathfrak{S}'$ of $\mathfrak{S}$.*

Proof We first show that every cell has a unique parent. Without loss of generality, we assume that $\mathfrak{S}'$ is the top left child of $\mathfrak{S}$. We partition $\mathfrak{S}'$ into two regions A and B where the latter is the lower right quarter of $\mathfrak{S}'$. Region B only contains the $4s$ innermost rings of $\mathcal{M}[\mathfrak{S}]$, since the rings of the γth group have side length $2^{\gamma-1}S/m = S/16s$ and thus $4s$ rings have width $S/4$. This is also illustrated in Fig. 3 for $s = 1$. Thus, every $\mathfrak{C} \in \mathcal{M}[\mathfrak{S}]$ is either completely contained in A or in B. The same is, by definition true for all $\mathfrak{C}' \in \mathcal{M}[\mathfrak{S}']$. Note that this already implies the moreover-part of the lemma.

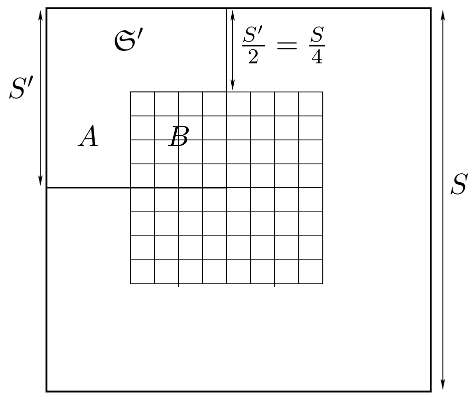


Fig. 3 The partition of $\mathfrak{S}'$ into regions A and B

Let $\rho(i)$ denote the distance of the i^{th} group from the boundary of $\mathfrak{S}$. By construction of the (s,t)-map, we have

$$\rho(i) = \sum_{k=1}^{i-1} 4s2^{k-1}\frac{S}{m} + \begin{cases} 0, & i \le 3, \\ 4s\frac{S}{m}, & i > 3 \end{cases}$$

and therefore we obtain

$$\rho(i) = \begin{cases} 4s(2^{i-1}-1)\frac{S}{m}, & i \le 3, \\ 4s2^{i-1}\frac{S}{m}, & i > 3. \end{cases}$$

Note that $\rho(i)$ does not depend on s, as $m = 8s2^\tau$. Moreover, let $\delta(i,j)$ denote the distance of the jth ring in the ith group from the boundary of $\mathfrak{S}$. We have

$$\delta(i,j) = \rho(i) + (j-1)2^{i-1}\frac{S}{m}.$$

Similarly, we define $\rho'(i)$ and $\delta'(i,j)$ for $\mathfrak{S}'$. Note that the side length of $\mathfrak{S}'$ is $S' = S/2$.

We first show that every cell of $\mathcal{M}[\mathfrak{S}']$ in region A has an unique parent in $\mathcal{M}[\mathfrak{S}]$. That is, we need to show that there exists (i',j') for all (i,j) such that $\delta'(i',j') = \delta(i,j)$. If $i > 3$ then $\rho'(i+1) = \rho(i)$ and thus

$$\delta(i,j) = \rho(i) + (j-1)2^{i-1}\frac{S}{m} = \rho(i-1) + (j-1)2^{i}\frac{S}{2m} = \delta'(i+1,j).$$

The case $i \le 3$ can be easily seen from Fig. 4 which illustrates the distances to the boundary of $\mathfrak{S}$ of the rings in the first three groups of $\mathcal{M}[\mathfrak{S}]$ (top) and those of the rings in the first four groups of $\mathcal{M}[\mathfrak{S}']$ (below) for the case $s = 1$. Note that choosing $s > 1$ increases the granularity, but the situation remains the same.

Now we consider region B. We have to show that there exists for all $1 \le j < 4s$ a (i',j') such that

$$\delta'(i',j') = \frac{S}{16s}j.$$

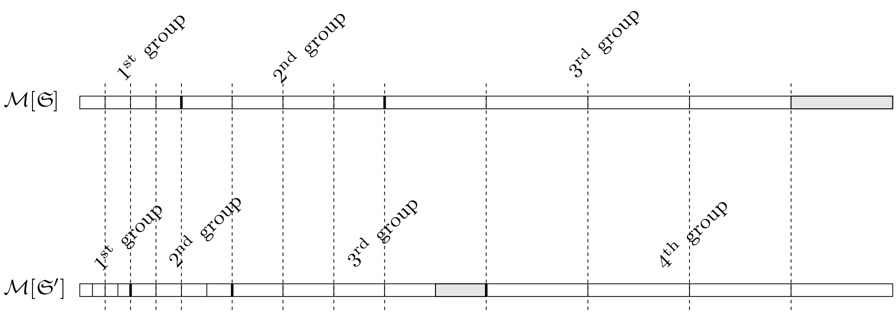


Fig. 4 The rings in the outermost groups of $\mathcal{M}[\mathfrak{S}]$ (*top*) and $\mathcal{M}[\mathfrak{S}']$ (*below*) for the case $s = 1$. The additional ring in third groups drawn *shaded*

For fixed $1 \le j < 4s$ choose $k \in \mathbb{Z}$ such that $4s \le j2^k < 8s$. Note that $k \ge 1$ and $k \le \log(8s) \ll \tau$. Let $i' = \tau - k + 1$ and observe that due to $\tau = \Theta(\log\log n)$ we may assume without loss of generality that $i' \ge 4$. Furthermore, let $j' = 1 + 2^{\tau - i' + 1} j - 4s$. Note that the choice of i' implies that $1 \le j \le 4s$. Now we easily check that

$$\begin{aligned}\delta'(i', j') &= 4s2^{i'-1}\frac{S}{2m} + (j'-1)2^{i'-1}\frac{S}{2m} = (4s + j' - 1)2^{i'-1}\frac{S}{2m}\\ &= (4s + j' - 1)2^{i'-1}\frac{S}{16s2^{\tau}} = (4s + j' - 1)2^{i'-\tau-1}\frac{S}{16s} = \frac{S}{16s}j,\end{aligned}$$

as desired. $\square$

5.2 Standardized Solutions

In this section, we will consider trees that enjoy certain structural properties. Roughly speaking, our PTAS optimizes over all such trees. The main result of this section is Theorem 10 which states that there exists an almost optimal tree having this structure. Let a and b be fixed and let $QT_{a,b}$ denote the corresponding shifted quadtree with (s, t)-maps on all its squares. Furthermore, let $\mathcal{ST}$ be an arbitrary Steiner tree. An *internal component* of $\mathcal{ST}$ with respect to $\mathfrak{S} \in QT_{a,b}$ is a connected component in the induced Steiner forest $\mathcal{ST}[V(\mathcal{ST}) \cap \mathfrak{S}]$. Let $k(\mathfrak{S}, \mathcal{ST})$ count the number of internal components of $\mathfrak{S}$. An edge $\{u, v\} \in \mathcal{ST}$ is an *external edge of* $\mathfrak{S}$, if exactly one endpoint of $\{u, v\}$ is contained in $\mathfrak{S}$. Let $\mathcal{E}(\mathfrak{S}, \mathcal{ST})$ denote the set of endpoints (within $\mathfrak{S}$) of external edges. An edge $\{u, v\} \in \mathcal{ST}$ has *level* l or *appears at level* l, if l is the least level, where $\{u, v\}$ is external. The level of $\{u, v\}$ is denoted by $\mathrm{lev}(\{u, v\})$.

Definition 9 Let $r \in \mathbb{N}$, $r \ge 2$. A Steiner tree $\mathcal{ST}$ is *(r, s)-standardized with respect to a and b* if every square $\mathfrak{S} \in QT_{a,b}$ is *(r, s)-standardized*, i.e., $\mathfrak{S}$ satisfies

(S1) $k(\mathfrak{S}, \mathcal{ST}) \le r$.
(S2) For all $\mathfrak{C} \in \mathcal{M}[\mathfrak{S}]$, all points in $\mathcal{E}(\mathfrak{S}, \mathcal{ST}) \cap \mathfrak{C}$ belong to the same internal component of $\mathfrak{S}$.

Henceforth we need the following technical concepts. Let $\mathbf{S}^*$ denote the set of all Steiner trees which have the same point set as $\mathcal{ST}^*$, i.e., $\mathcal{ST} \in \mathbf{S}^*$ if and only if $V(\mathcal{ST}) = V(\mathcal{ST}^*)$ and $S(\mathcal{ST}) = S(\mathcal{ST}^*)$. For fixed a and b, we define a transitive relation $\lhd$ on $\mathbf{S}^*$ as follows. For $\mathcal{ST}, \mathcal{ST}' \in \mathbf{S}^*$ we have $\mathcal{ST} \lhd \mathcal{ST}'$ if and only if there exists a bijection $\gamma : E(\mathcal{ST}) \to E(\mathcal{ST}')$ such that $\mathrm{lev}(e) \le \mathrm{lev}(\gamma(e))$ for all $e \in E(\mathcal{ST})$. We remind the reader that the dissection lines subdividing $\mathfrak{S}_0$ have level 0, and thus a small $\mathrm{lev}(e)$ means that e is cut near to the root.

Theorem 10 *For all $a, b \in \{0, 2, 4, \dots, L-2\}$ there exists an (r, s)-standardized Steiner tree $\mathcal{ST}^*_{a,b} \in \mathbf{S}^*$ with $\mathcal{ST}^* \lhd \mathcal{ST}^*_{a,b}$ such that if a and b are chosen uniformly at random then*

$$\mathbb{E}\left[\mathrm{val}\left(\mathcal{ST}^*_{a,b}\right) - \mathrm{val}\left(\mathcal{ST}^*\right)\right] \le \mathcal{O}\left(\frac{1}{s} + \frac{1}{\sqrt{r}}\right)\mathrm{val}\left(\mathcal{ST}^*\right).$$

We need several preliminary lemmas for the proof of Theorem 10. We start with the following simple statement which also appears implicitly in [3] and [6].

Lemma 11 *Let a and b be chosen uniformly at random from $\{0, 2, \ldots, L-2\}$ and let $\mathcal{ST}$ be some fixed Steiner tree. Furthermore, let $\mathcal{E}_{e,l}$ denote the event that $e \in \mathcal{ST}$ is an external edge at level l. Then*

$$\Pr\{\mathcal{E}_{e,l}\} \le \frac{2^{l+2}}{L} \ell_q(e).$$

Proof Let $\mathcal{E}'_{e,l}$ denote the event that e crosses a horizontal or vertical line which has level l in $QT_{a,b}$. Clearly, we have

$$\Pr\{\mathcal{E}_{e,l}\} = \Pr\left\{\bigcup_{j=0}^{l} \mathcal{E}'_{e,j}\right\} \le \sum_{j=0}^{l} \Pr\left\{\mathcal{E}'_{e,j}\right\}.$$

If the edge e has length $\ell_q(e)$, it crosses at most $\ell_q(e)/2$ horizontal dissection lines and $\ell_q(e)/2$ vertical ones. Recall, that dissection lines have even integral coordinates. Therefore, we have $L/2$ horizontal (vertical) dissection lines. Since there are 2^j dissection lines at level j, the probability that a fixed dissection line has level j is at most $2^{j+1}/L$. Note that this only an upper bound, as for instance dissections lines outside the (scaled) bounding box can never have level 0. Since e crosses at most $\ell_q(e)$ dissection lines, we obtain

$$\Pr\{\mathcal{E}_{e,l}\} \le \sum_{j=0}^{l} 2\frac{2^j}{L}\ell_q(e) = \frac{2^{l+2}}{L}\ell_q(e).$$

This completes the proof. □

Lemma 12 *Let $a, b \in \{0, 2, 4, \ldots, L-2\}$ and let $\mathcal{ST}'$, $\mathcal{ST}$ be Steiner trees such that $\mathcal{ST}' \lhd \mathcal{ST}$. Furthermore let $x_l(\mathcal{ST}')$ and $x_l(\mathcal{ST})$ count the number of edges in $\mathcal{ST}'$ and $\mathcal{ST}$, respectively that are external at level l. Then*

$$x_l\left(\mathcal{ST}'\right) \ge x_l(\mathcal{ST})$$

for all $0 \le l \le t$.

Proof Note that every edge which is external at level l is also external at all levels below l, i.e., at all levels l' with $l' > l$. Since $\mathcal{ST}' \lhd \mathcal{ST}$, there is a bijection γ from $E(\mathcal{ST}')$ to $E(\mathcal{ST})$ such that $\mathrm{lev}(e') \le \mathrm{lev}(\gamma(e'))$ for all $e' \in E(\mathcal{ST}')$. If $\gamma(e')$ is external at level l, then e' is also external at level l. Thus, if $\gamma(e')$ contributes to $x_l(\mathcal{ST})$ then e' contributes to $x_l(\mathcal{ST}')$. □

Lemma 13 *For all $a, b \in \{0, 2, 4, \ldots, L-2\}$ there exists a Steiner tree $\mathcal{ST}_{a,b} \in \mathbf{S}^*$ satisfying* (S1) *such that $\mathcal{ST}^* \lhd \mathcal{ST}_{a,b}$. Moreover, if a and b are chosen uniformly at* *random we have*

$$\mathbb{E}\big[\ell_q\left(\mathcal{ST}_{a,b}\right) - \ell_q\left(\mathcal{ST}^*\right)\big] \le \frac{9}{\sqrt{r}}\,\ell_q\left(\mathcal{ST}^*\right).$$

Proof Assume for the moment that a and b are fixed. In the sequel, we will construct a sequence of Steiner trees $\mathcal{ST}_t, \mathcal{ST}_{t-1}, \dots, \mathcal{ST}_0$ such that $\mathcal{ST}_t = \mathcal{ST}^*$ and such that $\mathcal{ST}_{a,b} := \mathcal{ST}_0$ has the desired properties. As in Sect. 4, let H and V denote horizontal and vertical dissection lines of the shifted quadtree. By definition, the dissection lines have even coordinates. Furthermore, let $\pi(H)$ and $\pi(V)$ count how often $\mathcal{ST}^*$ crosses the dissection lines H and V, respectively. One easily checks that

$$\sum_H \pi(H) + \sum_V \pi(V) \le \ell_q\left(\mathcal{ST}^*\right), \tag{3}$$

where the sums are taken over all horizontal (vertical) dissection lines of the shifted quadtree.

We proceed in bottom-up fashion, i.e., we start with $\mathcal{ST}_t = \mathcal{ST}^*$ and transform $\mathcal{ST}_l$ into $\mathcal{ST}_{l-1}$. In the leaves of $QT_{a,b}$ there is nothing to do, since every leaf contains at most one point or Steiner point. At level l we check for every square whether it satisfies (S1). If this is not the case, we perform some corrections to the tree. Thereafter every square at level l will satisfy (S1) and we move to level $l-1$.

Whenever we find a square $\mathfrak{S}$ at the current level, say l, which contains more than r internal components, we modify its interior as follows. Let $\mathcal{K}_1, \dots, \mathcal{K}_{k(\mathfrak{S},\mathcal{ST})=k}$ denote the internal components of $\mathfrak{S}$. For every such component, we choose a representative point $u_i \in \mathcal{K}_i$. We connect $u_1, \dots, u_k$ by a Hamiltonian path and remove $k-1$ external edges. We call this modification a *reduction operation* (Fig. 5). It is well-known [15, 25] that the shortest salesman tour through k points in a square of side length S has length at most $S\sqrt{k}$. Thus we can find a Hamiltonian path which has length at most $S\sqrt{k}$, where $S = L/2^l$ is the side length of $\mathfrak{S}$. At level l, we can

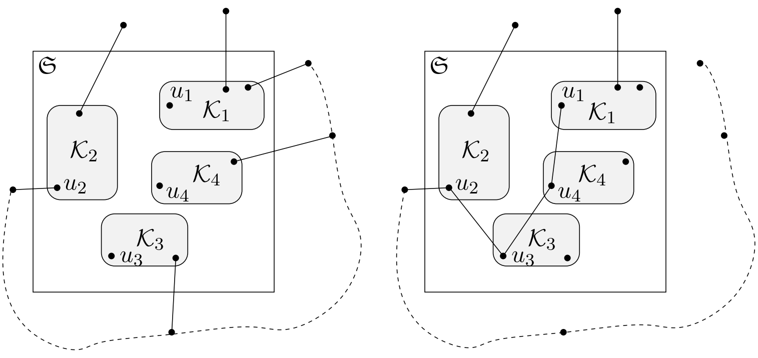


Fig. 5 A reduction operation on $\mathfrak{S}$ for $r \le 3$. Adding a Hamiltonian path (*right*) of length k allows us to remove $k-1$ external edges and reduces the number of internal components to 1

thus charge to every external edge which we delete a cost of

$$\delta_l := \frac{1}{k-1} \cdot \frac{L\sqrt{k}}{2^l} \le \frac{2L}{2^l\sqrt{k}} \le \frac{2L}{2^l\sqrt{r}},$$

since $k \ge r \ge 2$. Now, (S1) is satisfied in $\mathfrak{S}$ and we proceed similarly for the remaining squares at level l.

Since the procedure above does not affect the point set of the tree, we have $\mathcal{ST}_{a,b} \in \mathbf{S}^*$. Note, that the edges we add have always greater level than those we delete. We obtain the bijection $\gamma : E(\mathcal{ST}^*) \to E(\mathcal{ST}_{a,b})$ by defining an arbitrary bijection between the deleted external edges and those added within the square. Thus $\mathcal{ST}^* \lhd \mathcal{ST}_{a,b}$.

Next, let us bound the expected increase of length due to all reduction operations. Let $c(H,a,b,l)$ count how many edges crossing H are charged and deleted due to a reduction operation at level l. Clearly, we have $c(H,a,b,l) = 0$ if $l < \text{lev}(H,b)$. Otherwise, if $l \ge \text{lev}(H,b)$ then $c(H,a,b,l)$ is independent of b. This can be easily checked as follows. Recall, that if $\text{lev}(H,b) \le l$ then H is present at levels l through t. For fixed a and b, let $S^l_{a,b}$ denote the set of squares on level l. Note, that the boundaries of the squares in $S^l_{a,b}$ are horizontal and vertical dissection lines that have level l or less. Let H denote a horizontal line with $\text{lev}(H,b) = l$. Changing b to b' such that $\text{lev}(H,b') < \text{lev}(H,b)$ does not affect the dissection at levels l through t, since $S^l_{a,b'} = S^l_{a,b}$. Hence, $c(H,a,b',l) = c(H,a,b,l)$ for all $l \ge \text{lev}(H,b)$. Thus, there exist values $\tilde{c}(H,a,l)$, such that

$$c(H,a,b,l) = \begin{cases} \tilde{c}(H,a,l), & l \ge \text{lev}(H,b), \\ 0, & \text{otherwise}. \end{cases}$$

With respect to the random choice of a and b let the random variable $\Delta_H = \sum_l c(H,a,b,l)$ count the increase of length along a horizontal line H. The expectation of Δ_H can be bounded by

$$\begin{aligned} \mathbb{E}[\Delta_H] &= \frac{4}{L^2}\sum_a\sum_b\sum_{l=\text{lev}(H,b)}^{t} \frac{2L}{2^l\sqrt{r}}\tilde{c}(H,a,l) \\ &\le \frac{4}{L^2}\sum_a\sum_{l=0}^{t} |\{b : \text{lev}(H,b) \le l\}| \frac{2L}{2^l\sqrt{r}}\tilde{c}(H,a,l) \\ &\le \frac{16}{L\sqrt{r}}\sum_a\sum_{l=0}^{t}\tilde{c}(H,a,l), \end{aligned}$$

where the last inequality follows from Observation 5. Since every edge is only charged once, we have

$$\sum_{l=0}^{t}\tilde{c}(H,a,l) \le \pi(H)$$

and thus

$$\mathbb{E}[\Delta_H] \le \frac{16}{L\sqrt{r}} \sum_a \pi(H) = \frac{8}{\sqrt{r}} \pi(H).$$

Proceeding similarly for the vertical lines and by using (3) we bound the total expected increase in length which is at most

$$\begin{aligned}\mathbb{E}[\Delta] &\le \frac{8}{\sqrt{r}} \left(\sum_H \pi(H) + \sum_V \pi(V) \right) \\ &\le \frac{8}{\sqrt{r}} \ell_q \left(\mathcal{ST}^* \right).\end{aligned}$$

This completes the proof. □

Proof of Theorem 10 We first show for fixed a and b how to obtain a tree for which (S1) and (S2) holds. By Lemma 13 we obtain a Steiner tree $\mathcal{ST}_{a,b}$ which satisfies (S1). Furthermore, we know that $\mathcal{ST}_{a,b} \in \mathbf{S}^*$ and $\mathcal{ST}^* \lhd \mathcal{ST}_{a,b}$.

All we have to do, is to make sure that (S2) holds. Proceeding bottom-up, we transform $\mathcal{ST}_{a,b}$ into a standardized tree $\mathcal{ST}^*_{a,b}$. As in the proof of Lemma 13, let $\mathcal{ST}_l$ denote the tree obtained after processing level l, that is we start with $\mathcal{ST}_t = \mathcal{ST}_{a,b}$ and end with $\mathcal{ST}_0 = \mathcal{ST}^*_{a,b}$. Let $\mathfrak{S}$ be a square at level l. Consider a cell $\mathfrak{C} \in \mathcal{M}[\mathfrak{S}]$ with side length C. We have to make sure, that all points in $\mathcal{E}(\mathfrak{S}, \mathcal{ST}_l) \cap \mathfrak{C}$ belong to the same internal component of $\mathfrak{S}$. This is achieved as follows.

Let $u, v \in \mathcal{E}$ be two points which are contained in $\mathfrak{C}$ and belong to different internal components. Furthermore, let e_u and e_v be external edges with endpoints u and v, respectively. If adding $\{u, v\}$ closes a cycle which contains e_u or e_v, we keep $\{u, v\}$ and remove either e_u or e_v whichever lies on the cycle. We call this operation, which is illustrated in Fig. 6(a), a *replacement of* e_v. Note, that $\mathrm{lev}(\{u, v\}) \ge \mathrm{lev}(e_v)$. Otherwise, we redirect e_v from v to u as illustrated in Fig. 6(b). This operation is called *redirection of* e_v. Note, that a redirection does not change the level of an edge. Proceeding iteratively for all cells and squares at level l, we obtain $\mathcal{ST}_l$.

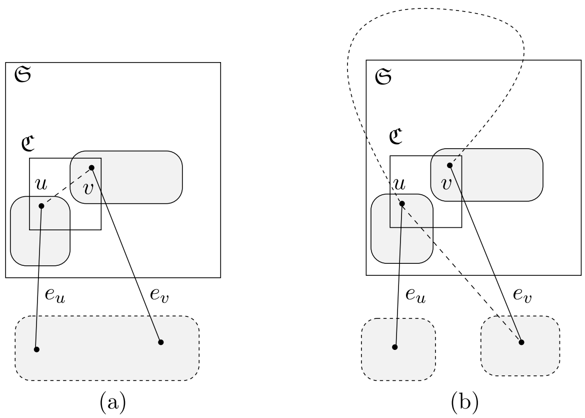


Fig. 6 Sketch for the proof of Theorem 10. If $\{u, v\}$ closes a cycle which contains e_u or e_v, then we can add $\{u, v\}$ to the tree an delete e_u or e_v instead (**a**). Otherwise, u and v are not connected over e_u and e_v. In this case we redirect e_v to u (**b**)

We have to check that replacements and redirections do neither violate (S2) on levels below l nor (S1) on *any* level. First, we show that (S1) is safe. This is quite obvious for the levels below l, since we reduce the number of internal components in the squares at level l. Consider a replacement which takes place in a square $\mathfrak{S}$ and let $\mathfrak{S}'$ denote any square containing $\mathfrak{S}$. The current tree induces a forest $\mathcal{F}'$ within $\mathfrak{S}'$. By inserting an edge $\{u, v\}$ we either close a cycle in $\mathcal{F}'$ or connect two of its components. In the first case the number of components does not change, since we only remove this cycle by deleting e_u or e_v, respectively. In the second case, the edge we delete might be completely contained in $\mathfrak{S}'$ and we obtain a new component. In any case, the number of components within $\mathfrak{S}'$ does not increase.

If we have a redirection in $\mathfrak{S}$ the situation is as follows. By construction, e_v is redirected. Let v' denote the external endpoint of e_v. If v' is not contained in $\mathfrak{S}'$ then the number of components within $\mathfrak{S}'$ does not change. Otherwise, e_v is contained in $\mathfrak{S}'$. In this case redirecting e_v does not change the number of components within $\mathfrak{S}'$, regardless whether u and v are connected within $\mathfrak{S}'$ or not. Hence, neither replacements nor redirections violate (S1) in any square of $\mathcal{QT}_{a,b}$.

Recall, that an edge which is external at level l is also external at levels $l+1$ through t. The replacement operation thus connects two endpoints of external edges. Since we proceed bottom-up, the cells on lower levels containing those endpoints are clean in a sense that they already fulfill (S2). Thus adding this edge will not violate (S2) on any level $l' > l$. In case of redirections we can argue similarly.

Both replacements and redirections delete one edge, say e, from the tree and add another edge e'. We call e' the *successor* of e. For every $e \in \mathcal{ST}_{a,b}$ we obtain a sequence

$$\mathcal{ST}_{a,b} \ni e \longrightarrow e' \longrightarrow e'' \longrightarrow \cdots \longrightarrow e_0 \in \mathcal{ST}^*_{a,b} \tag{4}$$

of successors. Every arrow stands for a single replacement or a single redirection. Observe, that the edge level is monotonically nondecreasing in this sequence. We denote the bijection which maps e to e_0 by γ. Clearly, $\mathrm{lev}(\gamma(e)) \geq \mathrm{lev}(e)$ and thus $\mathcal{ST}_{a,b} \lhd \mathcal{ST}^*_{a,b}$. Since $\lhd$ is transitive and $\mathcal{ST}^* \lhd \mathcal{ST}_{a,b}$, we have $\mathcal{ST}^* \lhd \mathcal{ST}^*_{a,b}$. Clearly, $\mathcal{ST}^*_{a,b} \in \mathbf{S}^*$, since we neither added Steiner points nor lost connectedness during the transformation process.

Finally, we bound the expected increase of length due to this transformation process if a and b are chosen at random. Recall, that for every edge $e \in \mathcal{ST}_{a,b}$ there is an edge sequence as given in (4). This sequence can be rewritten as a sequence $\sigma(e) = (\sigma_1, \ldots, \sigma_k)$ of redirections and replacements. One can easily check that σ_i is a redirection if $i < k$. In particular, $\sigma(e)$ contains at most one replacement which is then σ_k. If σ_k is a replacement, then the total cost of $\sigma(e)$ is bounded by the length of the edge we insert. This length is at most the length of the diagonal of the cell in which the new edge was placed.

Otherwise, if σ_k is a redirection, an upper bound can be obtained as follows. For every endpoint of $e = \{u, v\}$, we obtain a sequence of redirections $\sigma(u) = (\sigma^u_1, \ldots, \sigma^u_{k_u})$ and $\sigma(v) = (\sigma^v_1, \ldots, \sigma^v_{k_v})$, respectively. Recall, that by Lemma 8 every cell $\mathfrak{C}$ at level l has a unique parent $\mathfrak{C}'$ at level $l-1$. Thus, $\mathfrak{C} \cap \mathfrak{C}' \neq \emptyset$ implies $\mathfrak{C} \subseteq \mathfrak{C}'$. If $e = \{u, v\}$ is redirected in $\mathfrak{C}$ to $\{u', v\}$ and again in $\mathfrak{C}'$ to $\{u'', v\}$ then $\mathfrak{C}'$ contains both u and u'', since $u' \in \mathfrak{C} \cap \mathfrak{C}'$. Thus every cell where a redirection of e occurs contains the corresponding original endpoint, i.e., u and v, respectively. Thus, the total cost of $\sigma(u)$ can be bounded by the diagonal length of the cell where $\sigma^u_{k_u}$ takes place, since this cell also includes the original endpoint u. The increase of length due to $\sigma(e)$ can now be bounded as follows. Let $\sigma^u_{k_u}$ and $\sigma^v_{k_v}$ occur in cells $\mathfrak{C}_u$ and $\mathfrak{C}_v$, respectively.

Assume without loss of generality, that $\mathfrak{C}_u$ has greater side length. Then clearly, the cost of $\sigma(e)$ is bounded by twice the diagonal length of $\mathfrak{C}_u$. We partition $E(\mathcal{ST}_{a,b})$ into E_{out} and E_{in}. A cell which belongs to the first $4s$ rings of $\mathcal{M}[\mathfrak{S}]$ is called an *outer cell*. All other cells are *inner cells*. If σ_k is a replacement, then $e \in E_{\text{out}}$ if and only if the new edge is contained in an outer cell. Otherwise, if σ_k is a redirection then $e \in E_{\text{out}}$ if and only if $\mathfrak{C}_u$ (and thus also $\mathfrak{C}_v$) is an outer cell. Note, that E_{in} and E_{out} are random variables as they are defined with respect to $\mathcal{ST}_{a,b}$, which depends on the choice of a and b. The random variables Δ_{out} and Δ_{in} count the increase of length due to E_{out} and E_{in}, respectively.

First, we consider E_{out}. In this case it is convenient to estimate the error just roughly. As the side length of a square at level l is $L/2^l$ (cf. Observation 5), we deduce that the length of the diagonal of an *outer cell* in every $\mathcal{L}_q$-metric is at level l at most

$$\delta_l = \frac{L}{m} 2^{1-l}. \tag{5}$$

We charge at level l a cost of at most δ_l to $e \in E_{\text{out}}$ if we apply a replacement to e. Otherwise, if we apply a redirection we have a cost of at most δ_l per endpoint. Hence, we may assume that we charge at most $2\delta_l$ to e. That is we obtain the following rough upper bound:

$$\Delta_{\text{out}} \le \sum_{l=0}^{t} 2\delta_l \cdot x_l(\mathcal{ST}_{a,b}), \tag{6}$$

where the random variable $x_l(\mathcal{ST}_{a,b})$ counts the number edges in $\mathcal{ST}_{a,b}$ that are external at level l. Together with $\mathcal{ST}^* \lhd \mathcal{ST}_{a,b}$, Lemma 12 yields $x_l(\mathcal{ST}_{a,b}) \le x_l(\mathcal{ST}^*)$. Thus we obtain

$$\mathbb{E}\left[x_l(\mathcal{ST}_{a,b})\right] \le \mathbb{E}\left[x_l(\mathcal{ST}^*)\right] = \sum_{e \in E(\mathcal{ST}^*)} \Pr\left\{\mathcal{E}_{e,l}\right\}. \tag{7}$$

Thus by using Lemma 11 and $m = m(s) \ge 8st$, combining (6) and (7) yields

$$\begin{aligned}
\mathbb{E}[\Delta_{\text{out}}] &\le \sum_{l=0}^{t} \sum_{e \in E(\mathcal{ST}^*)} \Pr\left\{\mathcal{E}_{e,l}\right\} 2\delta_l \\
&\le \sum_{e \in E(\mathcal{ST}^*)} \sum_{l=0}^{t} \frac{2^{l+2}}{L} \ell_q(e) \frac{2L}{m} 2^{1-l} \\
&\le \sum_{e \in E(\mathcal{ST}^*)} \sum_{l=0}^{t} \frac{16}{m} \ell_q(e) = \frac{4}{s} \ell_q\left(\mathcal{ST}^*\right).
\end{aligned}$$

For $e \in E_{\text{in}}$ the situation is as follows. Observe, that by Lemma 7 the diagonal of an inner cell is less than its distance from the boundary of the enclosing square. Thus the length of the tree decreases if the last operation in $\sigma(e)$ is a replacement. Otherwise, let $\mathfrak{C}_u$ denote the largest cell in which a redirection takes place. By the considerations we made above, the total increase of length due to $\sigma(e)$ is at most twice the diagonal length of $\mathfrak{C}_u$. By Lemma 7 we know that this diagonal length is at most $\ell_q(e)/s$. That is, we have

$$\Delta_{\text{in}} \leq \sum_{e \in E_{\text{in}}} \frac{2\ell_q(e)}{s} \leq \sum_{e \in \mathcal{ST}_{a,b}} \frac{2\ell_q(e)}{s} \leq \frac{2}{s} \ell_q\left(\mathcal{ST}_{a,b}\right).$$

Finally, we bound the total expected increase of length, i.e., $\mathbb{E}[l_q(\mathcal{ST}) - l_q(\mathcal{ST}^*)]$. First, we estimate the expectation of $\Delta := \ell_q(\mathcal{ST}^*_{a,b}) - \ell_q(\mathcal{ST}_{a,b})$. By linearity of expectation and Lemma 13, we have

$$\begin{aligned}
\mathbb{E}[\Delta] &= \mathbb{E}[\Delta_{\text{out}}] + \mathbb{E}[\Delta_{\text{in}}] \\
&\leq \frac{4}{s} \ell_q\left(\mathcal{ST}^*\right) + \frac{2}{s} \mathbb{E}\left[\ell_q\left(\mathcal{ST}_{a,b}\right)\right] \\
&\leq \frac{4}{s} \ell_q\left(\mathcal{ST}^*\right) + \left(\frac{2}{s} + \frac{16}{s\sqrt{r}}\right) \ell_q\left(\mathcal{ST}^*\right)
\end{aligned}$$

and hence again using Lemma 13,

$$\begin{aligned}
\mathbb{E}\left[\ell_q\left(\mathcal{ST}^*_{a,b}\right) - \ell_q\left(\mathcal{ST}^*\right)\right] &\leq \mathbb{E}[\Delta] + \mathbb{E}\left[\ell_q\left(\mathcal{ST}_{a,b}\right) - \ell_q\left(\mathcal{ST}^*\right)\right] \\
&\leq \left(\frac{6}{s} + \frac{16}{s\sqrt{r}} + \frac{8}{\sqrt{r}}\right) \ell_q\left(\mathcal{ST}^*\right) \\
&= \mathcal{O}\left(\frac{1}{s} + \frac{1}{\sqrt{r}}\right) \ell_q\left(\mathcal{ST}^*\right).
\end{aligned}$$

Observe that $\mathcal{ST}^*_{a,b} = \mathcal{ST}_0$ spans the same points and Steiner points as $\mathcal{ST}^*$. Hence, we can bound $\text{val}(\mathcal{ST}^*_{a,b})$ as claimed. □

6 The Approximation Scheme

Although (r, s)-standardized trees have nice properties, we do not see an approach to compute an optimal one in polynomial time. Instead, our PTAS computes an almost optimal (r, s)-standardized tree. For fixed a and b, consider the optimal standardized Steiner tree $\mathcal{ST}^*_{a,b}$ and two squares $\mathfrak{S}_u$ and $\mathfrak{S}_v$ at level l. Assume that the two squares have a common parent $\mathfrak{S}$. Let $e = \{u, v\}$ be an edge that is contained within $\mathfrak{S}$ but has one endpoint in $\mathfrak{S}_u$ and the other in $\mathfrak{S}_v$, i.e, e appears at level l. There are cells $\mathfrak{C}_u$ and $\mathfrak{C}_v$ containing the endpoints of e in $\mathfrak{S}_u$ and $\mathfrak{S}_v$, respectively. The tree $\mathcal{ST}^*_{a,b}$ has the following structural property. All edges which appear at level l and have an endpoint in $\mathfrak{C}_u$ connect to the same internal component of $\mathfrak{S}_u$. The points in $\mathfrak{C}_u$ which belong to this component are from a structural point of view equivalent, since instead of connecting v to u we can connect v to any other of those points without adding a cycle or losing connectivity. Although the length of such an edge may be greater than $\ell_q(e)$, the absolute error is at most twice the side length of $\mathfrak{C}_u$. Of course, the same is true for $\mathfrak{C}_v$. Therefore, the distance between the centers of $\mathfrak{C}_u$ and $\mathfrak{C}_v$ is a good estimation to the length of e.

Under the assumption that this estimation suffices, it is quite obvious that the following information is sufficient to describe a standardized Steiner tree within $\mathfrak{S}$. Firstly, we specify in which cells we have endpoints of external edges. Intuitively speaking, such cells, later called portals, represent the component to which all external edges should be connected. Secondly, we encode which cells represent the same internal component of $\mathfrak{S}$. This motivates the following strategy for our PTAS. In a first phase we compute the "optimal" structure of our solution. This structure is described by edges between centers of cells and we minimize the total length of the cell-to-cell edges we use. In a second phase, those cell-to-cell edges are replaced by "real" edges.

6.1 Square Configuration

We describe the encoding in more detail. Let $\mathfrak{S}$ be a square in $QT_{a,b}$. The *configuration $\mathcal{C}(\mathfrak{S})$ of* $\mathfrak{S}$ is with respect to the irregular map $\mathcal{M}[\mathfrak{S}]$ and is given by the following parameters.

(1) For every cell $\mathfrak{C}$ we store a bit $\rho(\mathfrak{C})$ which indicates whether $\mathfrak{C}$ is a *portal cell*.
(2) For every cell $\mathfrak{C}$ we store a bit $\alpha(\mathfrak{C})$ which indicates whether $\mathfrak{C}$ is an *anchor cell*. We require, that every anchor cell is also a portal cell, i.e., $\alpha(\mathfrak{C}) \Rightarrow \rho(\mathfrak{C})$.
(3) A partition Z of the portal cells of $\mathfrak{S}$ into at most r sets.

The anchor bit is not required to encode the interior of a square but necessary to guarantee that our PTAS returns a connected tree. Intuitively speaking, $\rho(\mathfrak{C})$ indicates an obligation to connect to $\mathfrak{C}$ on some larger square. In this context the anchor bit indicates that this obligation is handed over to the next larger square.

Lemma 14 *The number of configurations for $\mathfrak{S}$ is at most $2^{\mathcal{O}(s^2 t \log r)}$.*

Proof By Lemma 6, $\mathcal{M}[\mathfrak{S}]$ contains at most $\mathcal{O}(s^2 t)$ cells. We store two bits per cell, $\rho(\mathfrak{C})$ and $\alpha(\mathfrak{C})$. Furthermore, the partition of the portal cells into at most r components can be encoded as a r-coloring of the cells. Altogether, we have

$$\left(2^{\mathcal{O}(s^2 t)}\right)^2 \cdot 2^{\mathcal{O}(s^2 t \log r)} = \mathcal{O}\left(2^{\mathcal{O}(s^2 t \log r)}\right)$$

configurations per rectangle. □

The configuration, where no bit is set is called the *empty configuration*. We write $\mathcal{C}(\mathfrak{S}) = \bigcirc$. The *configuration of $QT_{a,b}$*, denoted by $\mathcal{C}(QT_{a,b})$ is a set which contains tuples $\langle \mathfrak{S}, \mathcal{C}(\mathfrak{S}) \rangle$ representing the configurations of all the squares in $QT_{a,b}$ including $\mathfrak{S}_0$.

6.2 The Algorithm: First Phase

For every square $\mathfrak{S}$ and every configuration $\mathcal{C}(\mathfrak{S})$, we compute a value $T[\mathfrak{S}, \mathcal{C}(\mathfrak{S})]$ which can be seen as an almost tight upper bound to the cost of the optimal subtree within $\mathfrak{S}$ which has the structural properties specified by $\mathcal{C}(\mathfrak{S})$. Hence, $T[\mathfrak{S}_0, \bigcirc]$ is an upper bound to $\text{val}(\mathcal{ST}^*_{a,b})$. Since $\text{val}(\mathcal{ST}^*_{a,b})$ is a good upper bound to $\text{val}(\mathcal{ST}^*)$, it suffices to quantify the gap between $\text{val}(\mathcal{ST}^*_{a,b})$ and $T[\mathfrak{S}_0, \bigcirc]$. This will be done in Sect. 6.4.

If $\mathfrak{S}$ is a leaf of $QT_{a,b}$ then we have exactly one cell $\mathfrak{C}_0$ which contains a point with odd integral coordinates. This point need not be contained in $\mathscr{P}$. First we check whether $\rho(\mathfrak{C}) = 0$ for all $\mathfrak{C} \in \mathcal{M}[\mathfrak{S}] - \{\mathfrak{C}_0\}$. If this is not the case, we store $T[\mathfrak{S}, \mathcal{C}(\mathfrak{S})] = \infty$. Otherwise, we determine the penalty we pay in $\mathfrak{S}$. We have three cases. If $\rho(\mathfrak{C}_0) = 1$ and $\mathscr{P} \cap \mathfrak{C}_0 = \emptyset$ then we place a Steiner point into $\mathfrak{C}_0$ and set $T[\mathfrak{S}, \mathcal{C}(\mathfrak{S})] = c_S$. If $\rho(\mathfrak{C}_0) = 0$ and $\mathscr{P} \cap \mathfrak{C}_0 = \{u\}$ then we store $T[\mathfrak{S}, \mathcal{C}(\mathfrak{S})] = \pi(u)$. In all other cases, $T[\mathfrak{S}, \mathcal{C}(\mathfrak{S})] = 0$.

If $\mathfrak{S}$ is an internal node of $QT_{a,b}$ we proceed as follows. Assume that $\mathfrak{S}$ has level l and let $\mathfrak{S}_1, \dots, \mathfrak{S}_4$ denote its children. We enumerate all combinations of configurations for $\mathfrak{S}_1, \dots, \mathfrak{S}_4$. Let $\mathcal{C}(\mathfrak{S}_1), \dots, \mathcal{C}(\mathfrak{S}_4)$ be such a choice of configurations. For every square $\mathfrak{S}_i$, $\mathcal{C}(\mathfrak{S}_i)$ defines a partition $Z(\mathfrak{S}_i)$ of the portal cells in $\mathfrak{S}_i$ into at most r classes. We now construct a (not necessarily unique) forest $\mathcal{Z}(\mathfrak{S}_i)$ with the portal cells of $\mathfrak{S}_i$ as vertices such that $\mathcal{Z}(\mathfrak{S}_i)$ has the property that two portal cells $\mathfrak{C}_1$ and $\mathfrak{C}_2$ belong to the same connected component of $\mathcal{Z}(\mathfrak{S}_i)$ if and only if they are in the same partition of $Z(\mathfrak{S}_i)$. For the sake of brevity, we use $\mathcal{Z}' = \bigcup_{i=1}^{4} \mathcal{Z}(\mathfrak{S}_i)$. Let $\mathfrak{C}$ be a portal cell and let $\mathfrak{C}' \in \text{child}[\mathfrak{C}]$. If $\alpha(\mathfrak{C}') = 1$ then we say that $\mathfrak{C}'$ is an *hook cell* of $\mathfrak{C}$. Recall, that $\mathcal{Z}'$ is a graph on the portal cells of the children of $\mathfrak{S}$. A choice of configurations $\mathcal{C}(\mathfrak{S}_1), \dots, \mathcal{C}(\mathfrak{S}_4)$ is *admissible* for a configuration $\mathcal{C}(\mathfrak{S})$ if each of the following conditions holds.

(C1) Every portal cell in $\mathfrak{S}$ has at least one hook cell.
(C2) The parent in $\mathfrak{S}$ of every anchor cell in some $\mathfrak{S}_i$ is a portal cell, unless $\mathfrak{S} = \mathfrak{S}_0$.
(C3) All hook cells of a portal cell in $\mathfrak{S}$ belong to the same connected component of $\mathcal{Z}'$. Furthermore, there exists a graph $\mathcal{D}$ on the portal cells of $\bigcup_i \mathfrak{S}_i$ which has the following properties.
(C4) $\mathcal{D}$ uses only edges which cross the dissection lines that divide $\mathfrak{S}$ into its children.
(C5) Every connected component of $\mathcal{D} \cup \mathcal{Z}'$ contains at least one anchor cell, unless $\mathfrak{S} = \mathfrak{S}_0$.
(C6) Two portal cells $\mathfrak{C}_1$ and $\mathfrak{C}_2$ of $\mathfrak{S}$ belong to the same partition class of $Z(\mathfrak{S})$ if and only if their hook cells belong to the same connected component of $\mathcal{D} \cup \mathcal{Z}'$.

Remark 15 The relaxations of (C2) and (C5) are necessary to compute $T[\mathfrak{S}_0, \bigcirc]$. As the configuration $\bigcirc$ does not contain any portal cell at all, there would otherwise be no admissible configuration of the children.

The length of an edge in $\mathcal{D}$ is the distance of the centers of the portal cells which it connects in the $\mathcal{L}_q$-metric. Let $\ell_q(\mathcal{D})$ denote the total length of $\mathcal{D}$. If there is no combination which is admissible, we store ∞ in $T[\mathfrak{S}, \mathcal{C}(\mathfrak{S})]$. Otherwise, we choose admissible configurations $\mathcal{C}(\mathfrak{S}_1), \dots, \mathcal{C}(\mathfrak{S}_4)$ and a graph $\mathcal{D}$ such that

$$\sum_{i=1}^{4} T[\mathfrak{S}_i, \mathcal{C}(\mathfrak{S}_i)] + \ell_q(\mathcal{D}) + \sum_{\{\mathfrak{C}_1, \mathfrak{C}_2\} \in \mathcal{D}} [C(\mathfrak{C}_1) + C(\mathfrak{C}_2)] \tag{8}$$

is minimal, where $C(\mathfrak{C}_1)$ and $C(\mathfrak{C}_2)$ denote the side length of the portal cells $\mathfrak{C}_1$ and $\mathfrak{C}_2$, respectively. In essence, the third part (the sum over all edges in $\mathcal{D}$) of the formula above is necessary to bound the deviation in length from the optimal standardized solution. This will later be explained in more detail.

Given $\mathcal{C}(\mathfrak{S}_1), \dots, \mathcal{C}(\mathfrak{S}_4)$, we can use the following simple algorithm to compute the graph $\mathcal{D}$ for which

$$\ell_q(\mathcal{D}) + \sum_{\{\mathfrak{C}_1, \mathfrak{C}_2\} \in \mathcal{D}} [C(\mathfrak{C}_1) + C(\mathfrak{C}_2)]$$

is minimal. Let $\mathcal{K}_1, \dots, \mathcal{K}_k$ denote components of the $\mathcal{Z}'$. Let $\mathcal{B}$ denote a complete weighted graph on $\mathcal{K}_1, \dots, \mathcal{K}_k$. If $\mathcal{K}_i$ and $\mathcal{K}_j$ belong to the same square, then $\omega(\mathcal{K}_i, \mathcal{K}_j) = \infty$ due to (C4). Otherwise,

$$\omega(\mathcal{K}_i, \mathcal{K}_j) = \min_{c_1 \in V(\mathcal{K}_i), c_2 \in V(\mathcal{K}_j)} \left\{ d_q(c_1, c_2) + C(c_1) + C(c_2) \right\}, \tag{9}$$

where $d_q(c_1, c_2)$ denotes the distance between the centers of c_1 and c_2. To compute $\mathcal{D}$, it suffices to compute minimum spanning forest on $\mathcal{B}$ subject to (C5) and (C6). Due to (S1) we have at most r internal components in each $\mathfrak{S}_i$ and thus $k \le 4r$, i.e., the graph $\mathcal{B}$ has $4r$ vertices. As there are less than $(4r)^{4r}$ spanning forests on $\mathcal{B}$ (cf. [1]), we can find $\mathcal{D}$ by enumeration in time $\mathcal{O}((4r)^{4r})$.

It is quite obvious that we can find the configurations $\mathcal{C}^*(\mathfrak{S}_1), \dots, \mathcal{C}^*(\mathfrak{S}_4)$ that minimize (8) by exhaustive search. Let $\mathcal{D}^*_{\mathfrak{S}}$ denote the optimal graph $\mathcal{D}$ corresponding to $\mathcal{C}^*(\cdot)$. We compute $T[\mathfrak{S}_0, \bigcirc]$ by using a dynamic programming approach. If we proceed bottom-up, it suffices to read the $T[\mathfrak{S}_i, \mathcal{C}(\mathfrak{S}_i)]$ from the lookup table. We obtain $T[\mathfrak{S}_0, \bigcirc]$ which corresponds to some configuration $\mathcal{C}(QT_{a,b})$ that we have computed implicitly. One can easily check, that the running time of this dynamic program is

$$\begin{aligned}\left(2^{\mathcal{O}(s^2 t \log r)}\right)^5 \cdot \mathcal{O}\left((4r)^{4r}\right) \cdot \mathrm{poly}(n) &= 2^{\mathcal{O}(s^2 t \log r)} \cdot \mathcal{O}\left((4r)^{4r}\right) \cdot \mathrm{poly}(n) \\ &= \mathcal{O}\left((4r)^{4r} n^{\mathcal{O}(s^2 \cdot \log r)}\right).\end{aligned}$$

6.3 The Algorithm: Second Phase

Note that the first phase of our algorithm can be easily modified such that it not only returns $T[\mathfrak{S}_0, \bigcirc]$ but also a configuration for $QT_{a,b}$ corresponding to $T[\mathfrak{S}_0, \bigcirc]$ as well as a collection of graphs

$$D(\mathscr{P}) := \left\{ \mathcal{D}^*_{\mathfrak{S}} : \mathfrak{S} \in QT_{a,b} \right\}.$$

Intuitively speaking, this collection specifies which of the components are connected in which square. Even more, the edges to be used are roughly described by edges between centers of cells. During the second phase of our approximation scheme, we construct a Steiner tree $\mathcal{ST}$ from $D(\mathscr{P})$. This time, we proceed in top-down fashion. Let $\mathcal{ST} = \emptyset$. For some square $\mathfrak{S}$ at level l, we consider the graph $\mathcal{D}^*_{\mathfrak{S}}$. Let c be some edge in $\mathcal{D}^*_{\mathfrak{S}}$ which connects (the centers) of two cells $\mathfrak{C}_1$ and $\mathfrak{C}_2$. By recursively following the hook cells of both $\mathfrak{C}_1$ and $\mathfrak{C}_2$ we determine points u_1 and u_2 which serve as endpoints. In other words, starting at portal cell $\mathfrak{C}_1$ we choose one of its hook cells, then we choose an hook cell of this anchor cell, and so on. Thus, we go to lower and lower levels in $QT_{a,b}$ until we reach a leaf. In a leaf, we find by construction a point in $\mathscr{P}$ or Steiner point which we choose as u_1. We proceed similarly for $\mathfrak{C}_2$, add the edge $\{u_1, u_2\}$ to $\mathcal{ST}$ and mark c as done. As u_1 and u_2 are within $\mathfrak{C}_1$ and $\mathfrak{C}_2$, (8) implies

Lemma 16 $\ell_q(\mathcal{ST}) \leq T[\mathfrak{S}_0, \bigcirc]$.

In addition, we have to prove that $\mathcal{ST}$ is a Steiner tree on $V(\mathcal{ST}) \cup S(\mathcal{ST})$, that is, $\mathcal{ST}$ is cycle free and connected and spans all points in $V(\mathcal{ST}) \cup S(\mathcal{ST})$.

Lemma 17 *$\mathcal{ST}$ is a Steiner tree on $V(\mathcal{ST}) \cup S(\mathcal{ST})$.*

Proof From conditions (C1) and (C3) we obtain the following simple observation. Assume we follow recursively *all* the hook cells of some portal cell $\mathfrak{C}$ in a square $\mathfrak{S}$—not just one path as in the second phase of PTAS. Then the points we reach belong to the same internal component of $\mathfrak{S}$. We say that the portal cell $\mathfrak{C}$ *represents* those points. Thus, in the second phase of our PTAS, we may choose any of the points that are represented by $\mathfrak{C}$. Now, one can easily check that $\mathcal{ST}$ is cycle free. It remains to show, that $\mathcal{ST}$ is connected.

By (C3), (C5) and (C6) it is clear, that for every square $\mathfrak{S}$ the points represented by two portal cells $\mathfrak{C}$ and $\mathfrak{C}'$ in $\mathfrak{S}$ belong to the same internal component of $\mathfrak{S}$ if $\mathfrak{C}$ and $\mathfrak{C}'$ belong to the same connected component of $\mathcal{Z}(\mathfrak{S})$. We say that $\mathfrak{C}$ and $\mathfrak{C}'$ *represent* this component. Finally, the conditions (C2), (C5) and (C6) make sure that $\mathcal{ST}$ is connected. Assume $\mathcal{ST}$ is just a Steiner forest and let $\mathcal{K}$ be a connected component in $\mathcal{ST}$. Let l be the highest level where $\mathcal{K}$ was represented by some portal cell $\mathfrak{C}$. If $\alpha(\mathfrak{C}) = 1$ then by (C2) its parent is portal cell. Otherwise, by (C5) we would have connected $\mathcal{K}$ to some other component of $\mathcal{ST}$ which is by condition (C5) represented by some anchor cell. Thus, $\mathcal{K}$ is also represented at level $l - 1$ which is a contradiction. □

Note, that $\mathcal{ST}$ may in general be self-intersecting. It is well-known, that the length of a Steiner tree which intersects can be improved by replacing crossing edges. We can thus improve our solution in a last step.

6.4 Analysis

From Theorem 10, we obtain the following statement which yields together with Lemma 16 the approximation ratio of our algorithm.

Theorem 18 *There are* $a, b \in \{0, 2, 4 \ldots, L-2\}$ *such that*

$$T[\mathfrak{S}_0, \bigcirc] \le \mathcal{O}\left(1 + \frac{1}{s} + \frac{1}{\sqrt{r}}\right) \mathrm{val}\left(\mathcal{ST}^*\right).$$

Proof For a random choice of a and b, Theorem 10 guarantees that there is a standardized Steiner tree $\mathcal{ST}^*_{a,b}$ such that

$$\mathbb{E}\left[\mathrm{val}\left(\mathcal{ST}^*_{a,b}\right) - \mathrm{val}\left(\mathcal{ST}^*\right)\right] = \mathcal{O}\left(\frac{1}{s} + \frac{1}{\sqrt{r}}\right) \mathrm{val}\left(\mathcal{ST}^*\right), \tag{10}$$

$V(\mathcal{ST}^*_{a,b}) = V(\mathcal{ST}^*)$, $S(\mathcal{ST}^*_{a,b}) = S(\mathcal{ST}^*)$ and $\mathcal{ST}^* \lhd \mathcal{ST}^*_{a,b}$. We later construct a configuration of $QT_{a,b}$ from $\mathcal{ST}^*_{a,b}$ which proves that

$$\mathbb{E}\left[T[\mathfrak{S}_0, \bigcirc] - \mathrm{val}\left(\mathcal{ST}^*_{a,b}\right)\right] \le \frac{4}{s}\mathrm{val}\left(\mathcal{ST}^*\right) + \frac{4}{s}\mathbb{E}\left[\mathrm{val}\left(\mathcal{ST}^*_{a,b}\right)\right]. \tag{11}$$

Thus, we deduce from (10) that

$$\begin{aligned}\mathbb{E}\left[T[\mathfrak{S}_0, \bigcirc] - \mathrm{val}\left(\mathcal{ST}^*_{a,b}\right)\right] &\le \frac{4}{s}\mathrm{val}\left(\mathcal{ST}^*\right) + \frac{4}{s}\mathcal{O}\left(1 + \frac{1}{s} + \frac{1}{\sqrt{r}}\right) \mathrm{val}\left(\mathcal{ST}^*\right) \\ &= \mathcal{O}\left(\frac{1}{s} + \frac{1}{\sqrt{r}}\right) \mathrm{val}\left(\mathcal{ST}^*\right).\end{aligned}$$

Therefore both $\mathbb{E}[T[\mathfrak{S}_0, \bigcirc] - \mathrm{val}(\mathcal{ST}^*_{a,b})]$ and $\mathbb{E}[\mathrm{val}(\mathcal{ST}^*_{a,b}) - \mathrm{val}(\mathcal{ST}^*)]$ are $\mathcal{O}(1/s + 1/\sqrt{r})\mathrm{val}(\mathcal{ST}^*)$, and we finally have

$$\mathbb{E}\left[T[\mathfrak{S}_0, \bigcirc] - \mathrm{val}\left(\mathcal{ST}^*\right)\right] = \mathcal{O}\left(\frac{1}{s} + \frac{1}{\sqrt{r}}\right) \mathrm{val}\left(\mathcal{ST}^*\right).$$

Hence, there are $a, b \in \{0, 2, 4, \ldots, L-2\}$ such that

$$T[\mathfrak{S}_0, \bigcirc] \le \mathcal{O}\left(1 + \frac{1}{s} + \frac{1}{\sqrt{r}}\right) \mathrm{val}\left(\mathcal{ST}^*\right).$$

It remains to construct a configuration of $QT_{a,b}$ from $\mathcal{ST}^*_{a,b}$ which proves (11). Again, we proceed in bottom-up fashion. In the leaves of $QT_{a,b}$ we set the portal-bits of those cells which contain a point of $V(\mathcal{ST}^*_{a,b}) \cup S(\mathcal{ST}^*_{a,b})$. Then we move to level $t-1$. Let $\mathfrak{S}$ be a square at level l. Assume there is a cell $\mathfrak{C} \in \mathcal{M}[\mathfrak{S}]$ which contains a set $\mathcal{E}(\mathfrak{C})$ of endpoints of external edges. Since edges which are external at level l are also external at levels $l+1$ through t, all children of $\mathfrak{C}$ which contain points of $\mathcal{E}$ are already portal cells. We set the anchor-bit of all those portal cells and set $\rho(\mathfrak{C}) = 1$. Then clearly, both (C1) and (C2) hold. Note that we can derive the partition Z from the internal components of $\mathfrak{S}$. As $\mathcal{ST}^*_{a,b}$ is standardized, we deduce that (C3) holds.

Let $E(\mathfrak{S})$ denote the edges which appear at level $l+1$ and have both endpoints in $\mathfrak{S}$. All those edges cross at least one of the dissection lines which subdivide $\mathfrak{S}$ into its children. We construct $\mathcal{D}$ from $E(\mathfrak{S})$. Let $\{u_1, u_2\} \in E(\mathfrak{S})$ and let $\mathfrak{C}_1$ and $\mathfrak{C}_2$

denote the cells in the children of $\mathfrak{S}$ which contain u_1 and u_2, respectively. We add the edge connecting $\mathfrak{C}_1$ and $\mathfrak{C}_2$ to $\mathcal{D}$.

Recall that the length of edges in $\mathcal{D}$ is defined to be the distance between the cells' centers. This distance may be longer than $\{u_1, u_2\}$, but the absolute difference is at most $C_1 + C_2$, where C_1 and C_2 denote the side length of $\mathfrak{C}_1$ and $\mathfrak{C}_2$, respectively. In addition, we have to pay $C_1 + C_2$ for the third part of the sum in (8). Thus, this edge contributes at most $d_q(u_1, u_2) + 2C_1 + 2C_2$ to $T[\mathfrak{S}, \emptyset]$. Proceeding similarly for all edges in $E(\mathfrak{S})$, we obtain $\mathcal{D}$ which clearly satisfies (C4) through (C6).

Let $\Delta = |T[\mathfrak{S}, \bigcirc] - \ell_q(\mathcal{ST}^*_{a,b})|$ denote the total cost of this procedure. As in the proof of Theorem 10, we will split Δ into Δ_{in} and Δ_{out} such that $\Delta = \Delta_{\text{in}} + \Delta_{\text{out}}$. In order to bound Δ_{out} we use $\mathcal{ST}^* \lhd \mathcal{ST}^*_{a,b}$ to charge the increase of length to edges of the optimum. Let δ_l be defined as in (5). Thus we have to charge a cost of at most $2\delta_l$ per endpoint. The remainder of this proof mimics that of Theorem 10. Finally, we have

$$
\begin{aligned}
\mathbb{E}[\Delta_{\text{out}}] &\le \sum_{l=0}^{t} \sum_{e \in E(\mathcal{ST}^*)} \Pr\left\{\mathcal{E}_{e,l}\right\} 4\delta_l \\
&\le \sum_{e \in E(\mathcal{ST}^*)} \sum_{l=0}^{t} \frac{2^{l+1}}{L} \ell_q(e) \frac{4L}{m} 2^{1-l} \\
&\le \sum_{e \in E(\mathcal{ST}^*)} \sum_{l=0}^{t} \frac{16}{m} \ell_q(e) = \frac{4}{s} \ell_q\left(\mathcal{ST}^*\right).
\end{aligned}
$$

For all other edges, we simply have

$$
\Delta_{\text{in}} \le \sum_{e \in E_{\text{in}}} \frac{4\ell_q(e)}{s} \le \sum_{e \in \mathcal{ST}^*_{a,b}} \frac{4\ell_q(e)}{s} \le \frac{4}{s} \ell_q\left(\mathcal{ST}^*_{a,b}\right)
$$

by Lemma 7, and thus also

$$
\mathbb{E}[\Delta_{\text{in}}] \le \frac{4}{s} \mathbb{E}\left[\ell_q\left(\mathcal{ST}^*_{a,b}\right)\right].
$$

We obtain (11) due to $\Delta = \Delta_{\text{in}} + \Delta_{\text{out}}$. This completes the proof. □

Finally, using Theorem 10 and Lemma 16 one immediately obtains the following statement.

Corollary 19 $\ell_q(\mathcal{ST}) \le \mathcal{O}(1 + \frac{1}{s} + \frac{1}{\sqrt{r}})\text{val}(\mathcal{ST}^*)$.

Proof of Theorem 1 Recall, that the first phase of our PTAS has complexity $n^{\mathcal{O}(s^2 \cdot \log r)}$ whereas the running time of the second phase is clearly polynomial. Now, choosing $s = \mathcal{O}(1/\varepsilon)$ and $r = \mathcal{O}(1/\varepsilon^2)$ completes the proof. □

7 Higher Dimensions and Other Generalizations

A natural question is, whether there is also PTAS in higher fixed dimension. If $d \geq 3$ the number of cells in a (s,t)-map is $\Omega(t^{d-1})$ and thus the running time of our algorithm is no longer polynomial. However, we can construct a quasi-polynomial time approximation scheme. Instead of using (s,t)-maps it is more convenient to place a d-dimensional grid of granularity $\mathcal{O}(S\varepsilon/t)$ on a square of side length S. The remaining parts of our proofs have straight-forward equivalents, so one can show Theorem 2. Indeed, the whole analysis simplifies a lot if we use regular grids.

Our approach also extends to other tree problems. For example, assume that the input contains an additional points set $\mathcal{S}_0$ which restricts locations for Steiner points, i.e., every solution should satisfy $S(\mathcal{ST}) \subseteq \mathcal{S}_0$. This problem has a simple PTAS, if Steiner points have zero cost. It is then sufficient to add $\mathcal{S}_0$ to $\mathcal{P}$, to set the corresponding penalties to 0 and to chose $c_S = \infty$. If we assign (potential different) costs to the locations in $\mathcal{S}_0$, the problem still admits a PTAS which can be obtained by slightly modifying our algorithm. This is insofar surprising, as the corresponding network problem is not likely to be in $\mathcal{APX}$ [16].

References

1. Aigner, M., Ziegler, G.M.: Proofs from the Book, 3rd edn. Springer, New York (2004)
2. Arora, S.: Polynomial time approximation schemes for Euclidean TSP and other geometric problems. In: 37th Annual Symposium on Foundations of Computer Science, pp. 2–11 (1996)
3. Arora, S.: Polynomial time approximation schemes for Euclidean traveling salesman and other geometric problems. J. ACM **45**(5), 753–782 (1998)
4. Arora, S.: Approximation schemes for NP-hard geometric optimization problems: a survey. Math. Program. Ser. B **97**, 43–69 (2003)
5. Arora, S., Chang, K.: Approximation schemes for degree-restricted MST and red-blue separation problems. Algorithmica **40**(3), 189–210 (2004)
6. Arora, S., Raghavan, P., Rao, S.: Approximation schemes for Euclidean k-medians and related problems. In: Proceedings of the 30th Annual ACM Symposium on Theory of Computing, pp. 106–113 (1998)
7. Bern, M., Eppstein, D.: Approximation algorithms for geometric problems. In: Hochbaum, D. (ed.) Approximation Algorithms for NP-hard Problems. PWS Publishing (1997)
8. Bern, M., Plassmann, P.: The Steiner tree problem with edge length 1 and 2. Inf. Process. Lett. **32**, 171–176 (1989)
9. Courant, R., Robbins, H.: What is Mathematics? Oxford University Press, Oxford (1941)
10. Du, D.Z., Hwang, F.K.: A proof of Gilbert-Pollack's conjecture on the Steiner ratio. Algorithmica **7**, 121–135 (1992)
11. Garey, M., Graham, R., Johnson, D.: Some NP-complete geometric problems. In: Proceedings of the 8th Annual ACM Symposium on Theory of Computing, pp. 10–22 (1976)
12. Garey, M., Graham, R., Johnson, D.: The complexity of computing Steiner minimal trees. SIAM J. Appl. Math. **32**, 835–859 (1977)
13. Goemans, M.X., Williamson, D.P.: A general approximation technique for constrained forest problems. SIAM J. Comput. **24**(2), 296–317 (1995)
14. Hwang, F.K., Richards, D., Winter, P.: The Steiner Tree Problem. Annals of Discrete Mathematics, vol. 53. North-Holland, Amsterdam (1995)
15. Karloff, H.: How long can a Euclidean traveling salesman tour be? SIAM J. Discrete Math. **2**(1), 91–99 (1989)
16. Klein, P., Ravi, R.: A nearly best-possible approximation algorithm for node-weighted Steiner trees. J. Algorithms **19**, 104–115 (1995)

17. Kolliopoulos, S., Rao, S.: A nearly linear-time approximation scheme for the Euclidean k-median problem. In: Proceedings of the 7th Annual European Symposium on Algorithms, pp. 378–389 (1999)
18. Mitchell, J.S.B.: Guillotine subdivisions approximate polygonal subdivisions: a simple new method for the geometric k-MST problem. In: Proceedings of the 7th Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 402–408 (1996)
19. Mitchell, J.S.B.: Guillotine subdivisions approximate polygonal subdivisions: a simple polynomial-time approximation scheme for geometric TSP, k-MST and related problems. SIAM J. Comput. **28**(4), 1298–1309 (1999)
20. Moss, A., Rabani, Y.: Approximation algorithms for constrained node weighted Steiner tree problems. In: Proceedings of the 33rd Annual ACM Symposium on Theory of Computing, pp. 373–382 (2001)
21. Prömel, H.J., Steger, A.: The Steiner Tree Problem—A Tour through Graphs, Algorithms, and Complexity. Advanced Lectures in Mathematics. Vieweg, Braunschweig (2002)
22. Remy, J., Steger, A.: A quasi-polynomial time approximation scheme for minimum weight triangulation. In: Proceedings of the 38th Annual ACM Symposium on Theory of Computing, pp. 316–325 (2006)
23. Robins, G., Zelikovsky, A.: Improved Steiner tree approximation in graphs. In: Proceedings of the 11th Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 770–779 (2000)
24. Segev, A.: The node-weighted Steiner tree problem. Networks **17**(1), 1–17 (1987)
25. Supowit, J., Plaisted, D.A., Reingold, E.M.: The traveling salesman problem and minimum matching in the unit square. SIAM J. Comput. **12**(1), 144–156 (1983)
26. Talwar, K.: Bypassing the embedding: algorithms for low dimensional metrics. In: Proceedings of the 36th Annual ACM Symposium on Theory of Computing, pp. 281–290 (2004)
27. Trevisan, L.: When Hamming meets Euclid: the approximability of geometric TSP and MST. In: Proceedings of the 29th Annual ACM Symposium on Theory of Computing, pp. 21–29 (1997)