Rectilinear Trees under Rotation and Related Problems

[Extended Abstract]

Benny K. Nielsen
Dept. of Computer Science
University of Copenhagen
DK-2100 Copenhagen Ø
Denmark
benny@diku.dk

Pawel Winter
Dept. of Computer Science
University of Copenhagen
DK-2100 Copenhagen Ø
Denmark
pawel@diku.dk

Martin Zachariasen
Dept. of Computer Science
University of Copenhagen
DK-2100 Copenhagen Ø
Denmark
martinz@diku.dk

ABSTRACT

We consider the problem of finding a minimum spanning tree for a set of points in the plane where the orientations of edges are restricted to λ uniformly distributed orientations, $\lambda = 2, 3, 4, \ldots$, and where the coordinate system can be rotated around the origo by an arbitrary angle. The most important case with applications in VLSI design arises when $\lambda = 2$; in this, so-called rectilinear case, the edges have to be parallel with the x- or y-axis. We suggest a straightforward algorithm to solve this problem. We also discuss how to solve the rectilinear Steiner tree problem in the rotational setting. Finally, we provide some computational results indicating the average savings for different values of n and λ both for spanning and Steiner trees.

1. INTRODUCTION

Suppose that we are given a set P of n points in the plane. We are interested in finding a minimum spanning tree (MST) or a Steiner minimum tree (SMT) for P under the assumption that the edges are permitted to have a limited number of orientations. In fact, we will assume that these orientations are evenly spaced. More specifically, let λ be a non-negative integer, $\lambda \geq 2$. Let $\omega = \pi/\lambda$. Permissible directions are then defined by the rays initiating from the origo and making angles $i\omega$, $i = 0, 1, ..., 2\lambda - 1$ with the positive x-axis. Finding such an MST is not a challenging problem since it can be defined as a minimum spanning tree problem in a complete graph with appropriate edge lengths.

Suppose that we are permitted to rotate the coordinate system. Rotation by ω (or a multiple of ω) will have no impact on the lengths of the edges. But for any angle $\alpha \in [0, \omega[$, the edge lengths will change. What is the value of α minimizing the length of an MST for P? Once α is fixed, finding an MST is straightforward. Determining similar SMTs seems to be more complicated. However, for the most important rectilinear case, we will show that the search for such SMTs can be reduced to $O(n^2)$ rotational angles.

Our interest in rotated MSTs and SMTs with bounded orientations was motivated by recent developments in VLSI technology. It will soon be possible to manufacture chips with wires running in more than two directions. This makes the case $\lambda = 4$ important in practice. Furthermore, additional length savings seem to be available when the coordinate system can be rotated. This is in particular useful for small values of λ and for nets with limited number of terminals. As λ grows, the edge length variations become smaller. As the number of terminals increases, savings along some edges are "eaten" up by increased lengths of other edges.

2. LENGTH OF A SEGMENT

Let us first examine the situation where there are only two points $a = (a_x, a_y)$ and $b = (b_x, b_y)$. It is obvious that the best minimum spanning tree is obtained when the coordinate system is rotated so that the segment ab overlaps with one of the permissible orientations. So our problem is trivial. However, it is still interesting to determine how the length of ab changes as the coordinate system is rotated.

Assume that a and b are on the horizontal x-axis and $a_x < b_x$. When the coordinate system is rotated by α, $0 \leq \alpha \leq \omega$, then the length of ab, denoted by $|ab|_\alpha$, changes. It increases until $\alpha = \omega/2$, and then decreases until $\alpha = \omega$ (Fig. 1). More specifically,

$$|ab|_\alpha = |ab| \frac{\sin\alpha + \sin(\omega - \alpha)}{\sin(\omega)}$$

In particular, if $\omega = \pi/2$, then

$$|ab|_\alpha = |ab|(\sin\alpha + \cos\alpha)$$

It is obvious that the function $f_{ab}(\alpha) = |ab|_\alpha$ is periodically strictly concave (with the period ω). Furthermore, the minimum is attained when the direction of the segment ab overlaps with one of the direction rays.

3. MINIMUM SPANNING TREE

Consider a collection S of segments. Their total length will be minimized when the rotation of the coordinate system forces the orientation of one of the segments to overlap with one of the direction rays. This follows immediately from the fact that segment lengths are piecewise strictly concave

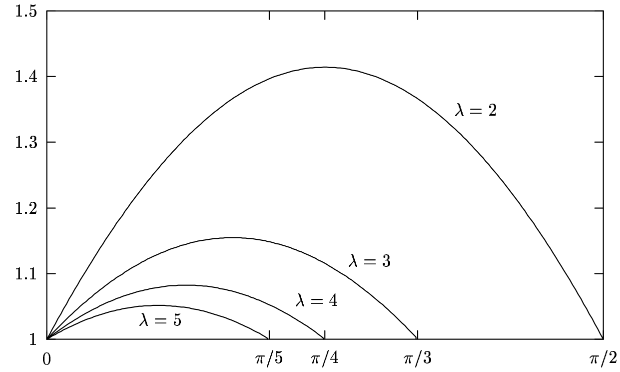

Figure 1: $f_{ab}(\alpha) = |ab|_\alpha$ for $\alpha \in [0, \frac{\pi}{\lambda}]$, $\lambda = 2, 3, 4, 5$.

functions of the rotation angle. The sum of piecewise strictly concave functions is also piecewise strictly concave.

Consider a set P of n points in the plane. Assume that the coordinate system has been rotated in such a way that an MST T for P is shortest possible. In view of the above remark, one of the edges of T overlaps with one of the direction rays.

The algorithm to determine T is therefore straightforward. For each pair of points a and b of P, consider the segment ab. Rotate the coordinate system so that the orientation of one of the direction rays overlaps with ab. Compute an MST for P and store it provided that it is shorter than any MST found so far. Fig. 2 shows how the lengths of MSTs for a set of 10 points and $\lambda = 4$ changes when the coordinate system is rotated. The histogram was generated by computing 1000 MSTs for values of α evenly distributed in the interval $[0, \frac{\pi}{4}]$. The vertical dashed lines indicate MSTs in fact computed by the algorithm. The optimum is on the x-axis.

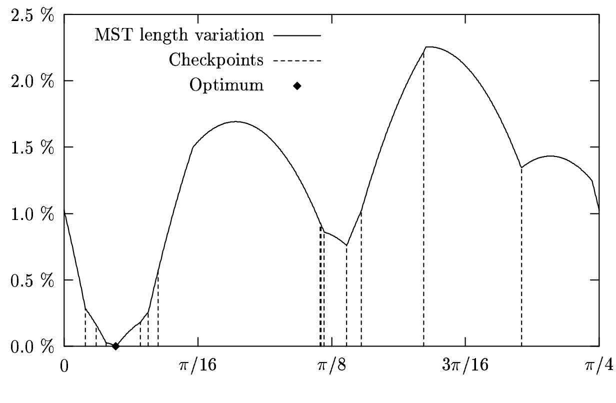

Figure 2: MST length variation.

For a fixed rotation of the coordinate system, an MST can be computed in $O(n \log n)$ time [6]. Since it is only necessary to consider $O(n^2)$ different rotations, the total running time for computing a rotational MST is $O(n^3 \log n)$. We will in Section 5 return to the issue of making this algorithm much more efficient in practice.

4. RECTILINEAR STEINER TREE

Consider the problem of interconnecting the set P of points by a tree of minimum total length, but *allowing* additional junctions, so-called Steiner points. When a set of λ, $\lambda \geq 2$, uniformly spaced legal orientations is given, this is denoted the Steiner tree problem in uniform orientation metrics [1, 4]. The rectilinear ($\lambda = 2$) and Euclidean ($\lambda = \infty$) Steiner tree problems have received considerable attention in the literature [3, 5].

In this section we consider the rotational variant of the rectilinear Steiner tree problem: Find a shortest interconnection of P using only two perpendicular orientations; note that the two orientations are not given, but are restricted to be perpendicular.

Assume that the coordinate system is rotated at an angle $0 \leq \alpha < \pi/2$. In this case the problem to be solved is the usual rectilinear Steiner tree problem, that is, to find a rectilinear Steiner minimum tree (RSMT). An RSMT is a union of full Steiner trees (FSTs) in which all terminals are leaves (and all interior nodes are Steiner points). Hwang [2] proved that there exists an RSMT for which every FST has a particular shape: The FST consists of a *backbone*, which is just a shortest path between two terminals, say a and b, using at most one vertical and at most one horizontal line segment (in Fig. 3 the backbone consists of the line segments ac and cb, where c is the corner point of the backbone); all the remaining terminals in the FST are connected directly to the backbone using exactly one line segment (e.g., segment ts in Fig. 3). Furthermore, none of the remaining terminals are connected to the backbone via the corner point of the backbone.

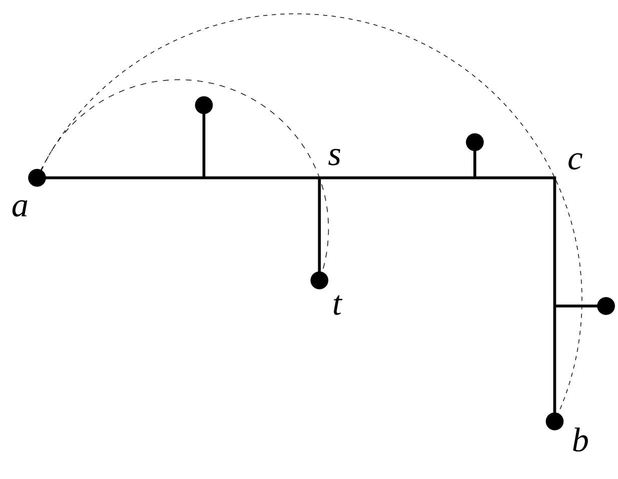

Figure 3: Rotating a rectilinear FST with one corner point.

Now let us assume that the rotational problem has an optimal solution for a given angle $0 \leq \alpha^* < \pi/2$. W.l.o.g. this optimal solution — which is an RSMT for the orientations given by α^* — consists of FSTs having the shape described above. We will now show that at least one of the FSTs in the RSMT must have a backbone that consists of *one* line segment, i.e., the backbone has no corner point.

Assume to the contrary that every FST has a backbone with a corner point. Let F be an FST and let $|F|_\alpha$ be the length of the FST as a function of α (for $\alpha^* - \epsilon < \alpha < \alpha^* + \epsilon$ where $\epsilon > 0$ is sufficiently small). Consider again the FST in Fig. 3. The corner point c will move along the indicated half-circle, such that the angle of the segments ac and cb remains at $\pi/2$. Thus the length of the backbone is a strictly concave function of α as shown in Section 2. Similarly, the length of a segment that connects one of the remaining terminals to the backbone will be a strictly concave function. Therefore, $|F|_\alpha$ is a strictly concave function of α, and the same will hold for the sum of all FST lengths. Therefore, the RSMT can in fact be shortened, which is a contradiction.

Since at least one of the FSTs must have a backbone without a corner point, the orientation of the backbone line segment will overlap with the orientation given by its two endpoints. The optimal solution to the rotational rectilinear Steiner tree problem can therefore be found using an algorithm similar to the one described in Section 3: For each pair of points a and b in P, rotate the coordinate system so that one of the direction rays overlaps with the segment ab. Compute an RSMT for this direction and repeat this procedure for all pairs of points; the shortest RSMT computed will be an optimal solution to the rotational rectilinear Steiner tree problem.

For $\lambda > 2$ we conjecture that a similar reduction of the necessary angles can be obtained, but it will not be polynomial in the number of terminals as for the rectilinear Steiner tree problem; it is not enough only to consider the orientations given by pairs of points in P.

5. COMPUTATIONAL RESULTS

In this section we give some computational results indicating the effect of allowing rotations of the coordinate system when computing MSTs and rectilinear SMTs. We used two sets of problem instances: VLSI design instances and randomly generated instances (uniformly distributed in a square). The VLSI instances were made available by courtesy of IBM and Research Institute for Discrete Mathematics, University of Bonn. In this study we have focused on one particular chip from 1996.

5.1 Minimum Spanning Tree

Generally most of the edges for a given point set will not be part of a MST for any rotation angle. It is a waste of time to 'straighten' these edges and compute MSTs. Luckily most of these edges can be pruned by using so-called bottleneck Steiner distances.

Consider the complete graph K_n where the vertices represent the points in the plane. Define the weight of the edge between vertices a and b to be $|ab|_{\omega/2}$. In other words, this weight is equal to the maximum distance between a and b when the coordinate system is rotated. Compute bottleneck Steiner distances in K_n. It is defined as the minimum of the longest edges encountered in all paths between a and b in K_n, and can be computed in time $O(n^2)$ for all pairs of points; see [3] for details. Let B_{ab} denote the bottleneck Steiner distance between a and b. Whenever $|ab| > B_{ab}$, then the edge ab cannot be in any minimum spanning tree generated during the rotation of the coordinate system. Fig. 4 indicates how many edges survive the pruning for $n = 50$ and $\lambda = 2, 3, 4, 5$.

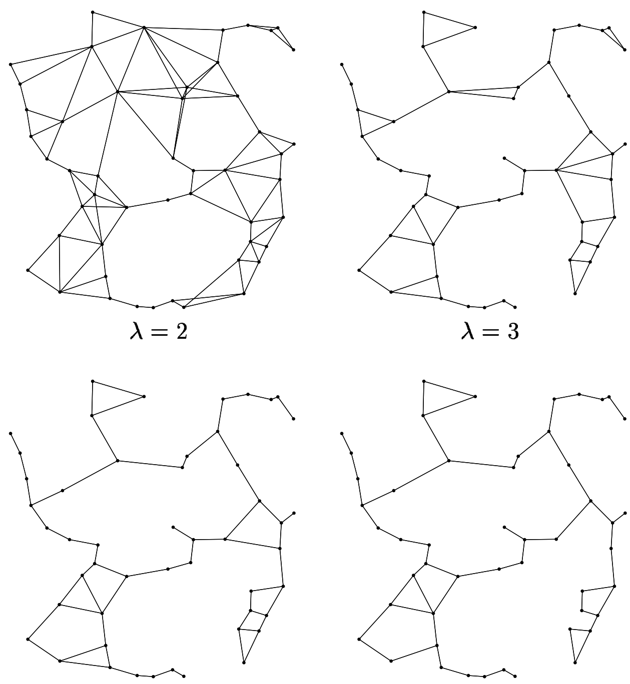

Figure 4: Edges surviving pruning of K_{50} for $\lambda = 2, 3, 4, 5$.

λ	n=3	n=4	n=5	n=10	n=20	n=50	n=100
2	2.76	4.86	7.38	20.12	44.77	122.09	253.84
3	2.34	3.84	5.50	12.62	26.77	71.43	146.81
4	2.22	3.44	4.77	10.77	23.05	60.51	123.05
5	2.13	3.34	4.54	10.17	21.28	56.11	113.59
6	2.07	3.27	4.38	9.83	20.54	53.89	108.57
7	2.06	3.21	4.28	9.54	20.05	52.50	105.88
8	2.03	3.12	4.18	9.36	19.76	51.74	104.49
9	2.02	3.09	4.14	9.32	19.64	51.09	103.29
10	2.02	3.07	4.10	9.29	19.56	50.69	102.49
16	2.01	3.03	4.03	9.08	19.17	49.73	100.33
32	2.00	3.00	4.00	9.01	19.03	49.14	99.41

Table 1: Number of surviving edges after pruning for randomly generated instances (averages over 100 runs).

This pruning turns out to be extremely efficient in general. Table 1 shows the number of edges that survive for different values of λ and n. Each entry is an average over 100 runs.

The extraordinary efficiency of pruning makes it possible to solve the problem of finding the best rotated MST much faster (especially for higher values of λ). It will on average require $O(n^2 \log n)$ time. However, it should be noted that it is possible to construct point sets of any size where the number of edges after pruning is $\Omega(n^2)$.

Table 2 shows the MST improvement for various values of n and λ where each entry is an average over 100 runs. The table contains two measures of improvement. The first one is calculated as $1 - \frac{|T_{\min}|}{|T_{\max}|}$ in percent. $T_{\min}$ is the shortest MST while $T_{\max}$ is the longest MST taken over 600 uniformly distributed values of α in the interval $[0, \omega[$. In the second measure of improvement the value $T_{\max}$ has been replaced by the length of the MST without rotating the points ($\alpha = 0$); a natural alternative.

5.2 Rectilinear Steiner Tree

We used GeoSteiner [5] to compute RSMTs for each of the $O(n^2)$ orientations given by the pairs of terminals (where n is the number of terminals). The RSMT improvement obtained by rotating the coordinate system can be seen in Table 3. The values are percentages which express the improvement compared to not rotating at all. Results for both random and VLSI instances are given. For small problem instances the improvements are highest for randomly generated instances, while for large instances, the VLSI problems result in a higher improvement.

6. RELATED AND OPEN PROBLEMS

There are several other geometric combinatorial optimization problems which require a selection of a subset of edges and can in the rotational setting be approached in the same way as the MST problem. The travelling salesman problem and matching are probably the most well-known. Another straightforward generalization occurs when the orientations are fixed but not necessarily evenly spaced. Determination of rotated MSTs in higher dimensions seems also to require a straightforward generalization of the 2-dimensional case.

There are several research directions which deserve more attention in the future. First of all, it remains open if the rotated MSTs can be determined more efficiently by some direct methods that do not involve enumeration of a (limited) number of MSTs. Also, the problem of determining rotated SMTs for other than the rectilinear case is completely open.

7. CONCLUDING REMARKS

We addressed the problem of determining MSTs and rectilinear SMTs when edge directions are limited to uniformly distributed orientations and where the coordinate system is permitted to rotate by any angle. We suggested a simple polynomial algorithm to solve the MST problem. We also provided some computational results indicating how big the savings can be. As it could be expected, the savings become negligible when λ and n grows. On the other hand, for all practical applications, λ is very small. Nets occurring in VLSI design are also rather small (in terms of the number of terminals involved). However, when many nets are to be routed, the overall savings will not be as impressive as for small isolated nets.

Acknowledgments

The authors would like to thank IBM and the Research Institute for Discrete Mathematics, University of Bonn, for providing industrial benchmarks circuits. The research was partially supported by the Danish Natural Science Research Council (contract number 56648).

8. REFERENCES

[1] M. Brazil, D. A. Thomas, and J. F. Weng. Minimum Networks in Uniform Orientation Metrics. *SIAM Journal on Computing*, 30:1579–1593, 2000.

[2] F. K. Hwang. On Steiner Minimal Trees with Rectilinear Distance. *SIAM Journal on Applied Mathematics*, 30:104–114, 1976.

[3] F. K. Hwang, D. S. Richards, and P. Winter. *The Steiner Tree Problem.* Annals of Discrete Mathematics 53. Elsevier Science Publishers, Netherlands, 1992.

[4] B. K. Nielsen, P. Winter, and M. Zachariasen. An Exact Algorithm for the Uniformly-Oriented Steiner Tree Problem. *Submitted*, 2001.

[5] D. M. Warme, P. Winter, and M. Zachariasen. GeoSteiner 3.1. Department of Computer Science, University of Copenhagen (DIKU), `http://www.diku.dk/geosteiner/`, 2001.

[6] P. Widmayer, Y. F. Wu, and C. K. Wong. On Some Distance Problems in Fixed Orientations. *SIAM Journal on Computing*, 16(4):728–746, 1987.

	$n=3$		$n=4$		$n=5$		$n=10$		$n=20$		$n=50$		$n=100$	
λ	$T_{\max}$	$\alpha=0$	$T_{\max}$	$\alpha=0$	$T_{\max}$	$\alpha=0$	$T_{\max}$	$\alpha=0$	$T_{\max}$	$\alpha=0$	$T_{\max}$	$\alpha=0$	$T_{\max}$	$\alpha=0$
2	21.86	14.48	17.42	9.67	15.86	8.38	10.29	4.89	7.47	3.19	5.03	2.12	3.46	1.51
3	8.86	5.34	7.38	4.34	6.37	3.69	4.31	2.52	3.01	1.47	1.70	0.78	1.23	0.61
4	5.55	3.48	4.32	2.50	3.81	2.14	2.37	1.31	1.72	0.93	1.02	0.51	0.72	0.39
5	3.40	2.10	2.85	1.66	2.42	1.39	1.59	0.89	1.09	0.58	0.72	0.39	0.47	0.25
6	2.36	1.46	1.88	1.17	1.55	0.97	1.05	0.67	0.73	0.39	0.44	0.21	0.30	0.16
7	1.76	1.12	1.41	0.86	1.21	0.74	0.78	0.42	0.57	0.32	0.35	0.19	0.24	0.14
8	1.36	0.85	1.11	0.66	0.94	0.53	0.61	0.33	0.44	0.24	0.25	0.13	0.18	0.09
9	1.04	0.66	0.80	0.49	0.73	0.43	0.49	0.28	0.30	0.16	0.20	0.11	0.15	0.08
10	0.88	0.52	0.70	0.40	0.58	0.31	0.40	0.21	0.26	0.13	0.18	0.09	0.11	0.05
16	0.32	0.20	0.26	0.16	0.22	0.14	0.16	0.10	0.11	0.06	0.07	0.03	0.04	0.02
32	0.08	0.05	0.07	0.04	0.06	0.04	0.04	0.02	0.03	0.01	0.02	0.01	0.01	0.01

Table 2: MST improvement in percent in relation to two values: The first one is the length of the worst case MST ($T_{\max}$) which is the longest MST found among 600 uniformly distributed values of α. The second one is the length of the MST when there is no rotation at all ($\alpha = 0$). All values are averages on 100 random instances.

	n=2	n=3	n=4	n=5	n=10	n=20	n=50	n=100
Random	21.21	11.46	8.19	7.00	3.17	2.17	1.28	0.92
VLSI	14.61	7.01	5.96	7.62	4.88	2.90	-	-

Table 3: RSMT improvement in percent obtained by rotating as compared to not rotating at all. Random: Randomly generated instances (averages over 100 runs). VLSI: VLSI design instances (averages over 100 runs).