A Better Lower Bound for Two-Circle Point Labeling

Alexander Wolff[1], Michael Thon[1], and Yinfeng Xu[2]

[1] Institute of Mathematics and Computer Science, Ernst Moritz Arndt University, Greifswald, Germany. awolff@mail.uni-greifswald.de, mithon42@gmx.de
[2] School of Management, Xi'an Jiaotong University, Xi'an, China. yfxu@xjtu.edu.cn

Abstract. Given a set P of n points in the plane, the two-circle point-labeling problem consists of placing $2n$ uniform, non-intersecting, maximum-size open circles such that each point touches exactly two circles.

It is known that it is NP-hard to approximate the label size beyond a factor of ≈ 0.7321. In this paper we improve the best previously known approximation factor from ≈ 0.51 to $2/3$. We keep the $O(n \log n)$ time and $O(n)$ space bounds of the previous algorithm.

As in the previous algorithm we label each point within its Voronoi cell. Unlike that algorithm we explicitely compute the Voronoi diagram, label each point *optimally* within its cell, compute the smallest label diameter over all points and finally shrink all labels to this size.

1 Introduction

Label placement is one of the key tasks in the process of information visualization. In diagrams, maps, technical or graph drawings, features like points, lines, and polygons must be labeled to convey information. The interest in algorithms that automate this task has increased with the advance in type-setting technology and the amount of information to be visualized. Due to the computational complexity of the label-placement problem, cartographers, graph drawers, and computational geometers have suggested numerous approaches, such as expert systems [AF84,DF89], zero-one integer programming [Zor90], approximation algorithms [FW91,DMM+97,WW97,ZP99], simulated annealing [CMS95] and force-driven algorithms [Hir82] to name only a few. An extensive bibliography about label placement can be found at [WS96]. The ACM Computational Geometry Impact Task Force report [C+96] denotes label placement as an important research area. Manually labeling a map is a tedious task that is estimated to take 50 % of total map production time [Mor80].

In this paper we deal with a relatively new variant of the general label placement problem, namely the two-label point-labeling problem. It is motivated by maps used for weather forecasts, where each city must be labeled with two labels that contain for example the city's name or logo and its predicted temperature or rainfall probability.

The two-label point-labeling problem is a variant of the one-label problem that allows sliding. Sliding labels can be attached to the point they label anywhere on their boundary. They were first considered by Hirsch [Hir82] who gave

D.T. Lee and S.-H. Teng (Eds.): ISAAC 2000, LNCS 1969, pp. 422–431, 2000.

an iterative algorithm that uses repelling forces between labels in order to eventually find a placement without or only few intersecting labels. Van Kreveld et al. gave a polynomial time approximation scheme and a fast factor-2 approximation algorithm for maximizing the number of points that are labeled by axis-parallel sliding rectangular labels of common height [vKSW99]. They also compared several sliding-label models with so-called fixed-position models where only a finite number of label positions per point is considered, usually a small constant like four [FW91,CMS95,WW97]. Sliding rectangular labels have also been considered for labeling rectilinear line segments [KSY99]. Another generalization was investigated in [DMM+97,ZQ00], namely arbitrarily oriented sliding labels.

Point labeling with circular labels, though not as relevant for real-world applications as rectangular labels, is a mathematically interesting problem. The one-label case has already been studied extensively [DMM+97,SW00,DMM00]. For maximizing the label size, the best approximation factor now is $\frac{1}{3.6}$ [DMM00].

The two- or rather multi-label labeling problem was first considered by Kakoulis and Tollis who presented two heuristics for labeling the nodes and edges of a graph drawing with several rectangles [KT98]. Their aim was to maximize the number of labeled features. The algorithms are based on their earlier work; one is iterative, while the other uses a maximum-cardinality bipartite matching algorithm that matches cliques of pairwise intersecting label positions with the elements of the graph drawing that are to be labeled. They do not give any runtime bounds or approximation factors.

For the problem that we consider in this paper, namely maximizing the size of circular labels, two per point, Zhu and Poon gave the first approximation algorithm [ZP99]. They achieved an approximation factor of $\frac{1}{2}$. Like all following algorithms, their algorithm relies on the fact that there is a region around each input point p such that p can be labeled within this region and this region does not intersect the region of any other input point. The size of the region—in their case a circle centered at p—and thus the labels is a constant fraction of an upper bound for the maximum label size.

Recently Qin et al. improved this result [QWXZ00]. They gave an approximation algorithm with a factor of $\frac{1}{1+\cos 18^\circ} \approx 0.5125$. They also state that it is NP-hard to approximate the label size beyond a factor of ≈ 0.7321. The regions into which they place the labels are the cells of the Voronoi diagram. However, they do not compute the Voronoi diagram explicitely, but use certain properties of its dual, the Delauney triangulation. For estimating the approximation factor of their algorithm they rely on the same upper bound for the maximum label size as Zhu and Poon, namely the minimum (Euclidean) distance of any pair of input points.

In this paper we give an algorithm that also places the labels of each point into its Voronoi cell. However, unlike the previous algorithm we do this optimally and compare the label diameter of our algorithm not to an upper bound but directly to the optimal label diameter. This yields an approximation factor of 2/3, which nearly closes the gap to the non-approximability result stated above.

At the same time we keep the $O(n \log n)$ time and $O(n)$ space bounds of the previous algorithms, where n is the number of points to be labeled.

This paper is organized as follows. In Section 2 we prove that in each cell of the Voronoi diagram of the given point set P there is enough space for a pair of uniform circular labels whose diameter is 2/3 times the optimal diameter for labeling P with circle pairs. In Section 3 we show how to label points optimally within their Voronoi cells and state our central theorem.

The actual label placement is a special case of a gift wrapping problem, where the gift is a coin (as large as possible) and the wrapping a convex polygonal piece of paper that can only be folded once along a line. Our problem is special in that it specifies a point on the folding line and thus takes away a degree of freedom. For the more general problem, the currently best known algorithm takes $O(m \log m)$ time where m is the number of vertices of the convex polygon [KKSY99]. For our special problem we have a linear-time algorithm, but due to space constraints we can only give an algorithm that uses standard techniques for computing the lower envelope of a set of "well-behaving functions". It runs in $O(m \log m)$ time.

Throughout this paper we consider labels being topologically *open*, and we define the *size* of a solution to be the diameter of the uniform circular labels. A label placement is *optimal* if no two labels intersect and labels have the largest possible size. We will only consider point sets of size $n \geq 2$.

2 The Lower Bound

The two-circle point-labeling problem is defined as follows.

Definition 1 (two-circle point-labeling problem). *Given a set P of n points in the plane, find a set of $2n$ uniform, non-intersecting, maximum-size open circles such that each point touches exactly two circles.*

Zhu and Poon [ZP99] have suggested the first approximation algorithm for the two-circle point-labeling problem. Their algorithm always finds a solution of at least half the optimal size. The algorithm is very simple; it relies on the fact that D_2, the minimum Euclidean distance between any two points of the input point set P is an upper bound for the optimal label size (i.e. diameter), see Figure 1. On the other hand, given two points p and q in P, open circles $C_{p,D_2/2}$ and $C_{q,D_2/2}$ with radius $\frac{1}{2}D_2$ centered at p and q do not intersect. Thus if each point is labeled within its circle, no two labels will intersect. Clearly this allows labels of maximum diameter $\frac{1}{2}D_2$, i.e. half the upper bound for the optimal label size.

The difficulty of the problem immediately comes into play when increasing the label diameter d beyond $\frac{1}{2}D_2$, since then the intersection graph of the (open) disks $C_{p,d}$ with radius d centered at points p in P changes abruptly; the maximum degree jumps from 0 to 6.

Recently Qin et al. have given an approximation algorithm that overcomes this difficulty and labels all points with circles slightly larger than the threshold

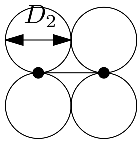

Fig. 1. D_2 is an upper bound for the optimal label size.

of $\frac{1}{2}D_2$. Their diameter is $d_{\text{old}} = \frac{1}{1+\cos 18°} D_2 \approx 0.5125\, D_2$. Their algorithm also assigns each point a certain region such that no two regions intersect and each point can be labeled within its region. The regions they use are not circles but the cells of the Voronoi diagram of P, a well-known multi-purpose geometrical data structure [Aur91]. Instead of computing the Voronoi diagram explicitely they use the dual of the Voronoi diagram, the Delauney triangulation $\mathrm{DT}(P)$ to apply a packing argument. $\mathrm{DT}(P)$ is a planar geometric graph with vertex set P and edges for each pair of points that can be placed on the boundary of an open disc that does not contain any other points of P [Aur91]. Qin et al. argue that in $\mathrm{DT}(P)$ each point p can have at most six short edges, where an edge pq is short if the Euclidean distance $d(p,q)$ of p and q is shorter than $2d_{\text{old}}$. They show that among the lines that go through these short edges there must be a pair of neighboring lines that forms an angle α of at least 36°. They place the circular labels of p with diameter d_{old} such that their centers lie on the angular bisector of α. Finally they prove that these labels lie completely within the Voronoi cell $\mathrm{Vor}(p)$ of p. The Voronoi cell of p is the (convex) set of all points in the plane that are closer to p than to any other input point. Thus the Voronoi cells of two different input points are disjoint and the labels of one input point cannot intersect the labels of any other.

Both the idea of our new algorithm and the proof of its approximation factor are simpler than those of its predecessor. However, in order to keep the $O(n \log n)$ time bound the implementation becomes slightly more involved. Our strategy is as follows. We first show that there is a region $Z_{\text{free}}(p)$ around each input point p that cannot contain any other input point. The size of $Z_{\text{free}}(p)$ only depends on the optimal label diameter. Let $Z_{\text{label}}(p)$ be the Voronoi cell of p that we would get if all points on the boundary of $Z_{\text{free}}(p)$ were input points. We can compute the largest label diameter d_{lower} such that two disjoint circular labels of p completely fit into $Z_{\text{label}}(p)$. We do not know the orientation of $Z_{\text{label}}(p)$ relative to p, but we know that $Z_{\text{label}}(p)$ lies completely in $\mathrm{Vor}(p)$. Thus our algorithm must go the other way round: we label each point optimally within its Voronoi cell, compute the smallest label diameter over all points and finally shrink all labels to this size. Then we know that each label is contained in the Voronoi cell of its point, and that the labels are at least as large as those that would have fit into $Z_{\text{label}}(p)$.

Let $\mathcal{C}_{\text{opt}}$ be a fixed optimal solution of the input point set P. $\mathcal{C}_{\text{opt}}$ can be specified by the label size d_{opt} and an angle $0 \le \alpha_p < 180°$ for each input point.

The angle α_p specifies the position of a line through p that contains the centers of the labels of p (at a distance $d_{\text{opt}}/2$ from p). By convention we measure α_p from the horizontal (oriented to the right) through p to the line (oriented upwards) through the label centers. In the following we assume that $d_{\text{opt}} = 1$; the input point set can always be scaled so that this is true.

Definition 2. *Let $C_{m,r}$ be an open disk with radius r centered at a point m and let H_{pq} be the open halfplane that contains p and is bounded by the perpendicular bisector of p and q. For each $p \in P$ let the* point-free zone $Z_{\text{free}}(p) = C_{Z_1,\frac{1}{2}\sqrt{3}} \cup C_{Z_2,\frac{1}{2}\sqrt{3}} \cup C_{p,1}$*, where Z_1 and Z_2 are the centers of the labels L_1 and L_2 of p in a fixed optimal solution $\mathcal{C}_{\text{opt}}$. The* label zone $Z_{\text{label}}(p)$ *is the intersection of all halfplanes H_{pq} with q a point on the boundary of $Z_{\text{free}}(p)$.*

Note that $Z_{\text{free}}(p)$ and $Z_{\text{label}}(p)$ are symmetric to the lines that form angles of α_p and $\alpha_p + 90°$ through p. The size of these areas only depends on d_{opt}, their orientation only on α_p, see Figure 2.

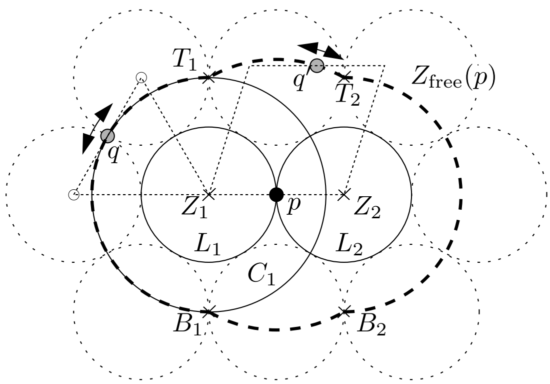

Fig. 2. The point-free zone does not contain any input point other than p.

Lemma 1. *$Z_{\text{free}}(p)$ does not contain any input points except p.*

Proof. Refer to Figure 2. It indicates that the boundary of $Z_{\text{free}}(p)$ is the locus of all input points that are as close to p as possible. First we will show that the two disks $C_i := C_{Z_i,\frac{1}{2}\sqrt{3}}$ $(i = 1, 2)$ do not contain any input point other than p.

Let T_i and B_i be the top- and bottommost points on C_i, respectively. The arc that forms the boundary of $Z_{\text{free}}(p)$ between the points B_1 and T_1 stems from rotating a potential input point q around the label L_1 of p in $\mathcal{C}_{\text{opt}}$ such that the labels of q both touch L_1. This means that the centers of these three labels of diameter 1 form an equilateral triangle whose height $\frac{1}{2}\sqrt{3}$ corresponds to the distance of q and Z_1. In other words, if there was a point $q \in P \setminus \{p\}$

with $d(q, Z_1) < \frac{1}{2}\sqrt{3} = \text{radius}(C_1)$, then in $\mathcal{C}_{\text{opt}}$ a label of q would intersect L_1, a contradiction to $\mathcal{C}_{\text{opt}}$ being an optimal solution. Due to symmetry the same holds for C_2. It remains to show that $C_{p,1}$, the third of the three circles that contributes to $Z_{\text{free}}(p)$, does not contain any input point other than p.

Consider the arc that forms the boundary of $Z_{\text{free}}(p)$ between the points T_1 and T_2. This arc is caused by rotating another potential input point q' around p such that each of the labels of q' touches a label of p. Observe that the centers of the four labels of q' and p form a rhombus of edge length 1 when q' moves continously from T_1 to T_2. Since q' and p are the midpoints of two opposite edges of the rhombus, their distance during this movement remains constant. Clearly the distance of the edge midpoints equals the edge length of the rhombus, i.e. 1, which in turn is the radius of $C_{p,1}$. Thus if q' entered the area $C_{p,1} \setminus (C_1 \cup C_2)$, a label of q' would intersect a label of p in $\mathcal{C}_{\text{opt}}$ —a contradiction. ⊕

Lemma 2. *The Voronoi cell of p contains the label zone $Z_{\text{label}}(p)$ of p.*

Proof. The Voronoi cell of p can be be written as $\text{Vor}(p) = \bigcap_{v \in P\setminus\{p\}} H_{pv}$. It contains $Z_{\text{label}}(p) = \bigcap_{v' \in \text{boundary}(Z_{\text{free}}(p))} H_{pv'}$ since for all input points $v \neq p$ there is a $v' \in \text{boundary}(Z_{\text{free}}(p))$ such that H_{pv} contains $H_{pv'}$. ⊕

Lemma 3. *For each input point p there are two disjoint circles of diameter $d_{\text{lower}} = \frac{2}{3}\, d_{\text{opt}}$ that touch p and lie completely within $Z_{\text{label}}(p)$.*

Proof. We do not compute the boundary of $Z_{\text{label}}(p)$ explicitely but parameterize the radius r of the labels of p such that we get the largest possible labels that do not touch the constraining halfplanes of $Z_{\text{label}}(p)$.

Our coordinate system is centered at p and uses four units for d_{opt}, see Figure 3. We show that for any point q on the boundary of $Z_{\text{free}}(p)$ the halfplane H_{pq} does not intersect the labels of p whose centers we place at $Z_1(-r, 0)$ and $Z_2(r, 0)$. The interesting case is that q lies on the circle C_2 and above the x-axis. Let q' be the point in the center of the line segment $\overline{pq}$. Then q' lies on the bold dashed circle with the equation $(x-1)^2 + y^2 = 3$. We parameterize the x-coordinate of q' using the variable $t \in [1, 1+\sqrt{3}]$. The vector $q' = (t, \sqrt{3-(t-1)^2})$ is normal to the boundary h_{pq} of the halfplane H_{pq}. Hence the distance of any point s to h_{pq} is given by the normalized scalar product of $(s - q')$ and q', namely $d(s, h_{pq}) = \frac{|(s-q')q'|}{\|q'\|}$. For $s = Z_2(r, 0)$, the center of one of the labels of p, we have

$$r = d(Z_2, h_{pq}) = \frac{|rt - 2t - 2|}{\sqrt{2(t+1)}}.$$

Since the enumerator is always less, and the denominator greater than zero, we get $r = f(t) = \frac{2(t+1)}{t+\sqrt{2(t+1)}}$. The zeros of f' are given by the equation $0 = \sqrt{2(t+1)} - 1 - \frac{(t+1)}{\sqrt{2(t+1)}}$ which yields a minimum of $r = \frac{4}{3}$ at $t = 1$ for $t \in [1, 1+\sqrt{3}]$. This value of r corresponds to a label diameter of $\frac{2}{3}$ according to the original scale with $d_{\text{opt}} = 1$. The case for $Z_1(-r, 0)$ is symmetric. ⊕

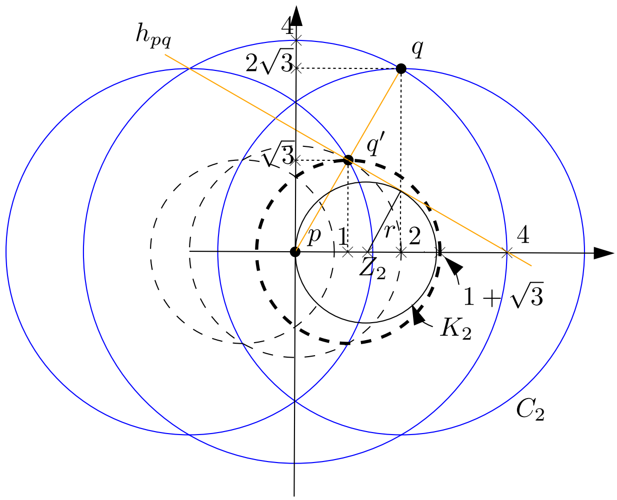

Fig. 3. Estimating the size of the label K_2 of p.

3 The Algorithm

In this section we first show how to label a point optimally within its Voronoi cell and then combine this result with the lower bound of the previous section to design a new approximation algorithm for two-circle point-labeling.

Lemma 4. *A point p that lies in the interior of a convex polygon G with m vertices $v_1, \ldots, v_m$ can be labeled with two disjoint uniform circles of maximum size in $O(m \log m)$ time and $O(m)$ space.*

Proof. Our algorithm works conceptually as follows. We compute a continuous function f that maps an angle $0 \le \alpha < 2\pi$ to the largest radius r_α such that the disk C_{z_α, r_α} touches p and lies completely within G, and such that at the same time the disk's center z_α, the point p and the horizontal $\overrightarrow{h_p}$ (directed to the right) through p form an angle of size α.

Since we want to find an angle that maximizes the common radius of *two* disjoint circular labels of p, we have to determine an angle α such that $\min(r_\alpha, r_{\alpha+\pi})$ is maximized. We do this by computing the maximum of the lower envelope of f and f' where $f'(\alpha) = f(\alpha + \pi \bmod 2\pi)$. Since f (and thus f', too) consists of at most $2n$ monotonous pieces (see below), the maximum value of the piecewise minimum of f and f' can be computed from a suitable representation of f in linear time. Finally the position of the centers of the maximum size labels of p is given by the angle α that yields the maximum value of $\min(f(\alpha), f'(\alpha))$. The labels' radius is of course r_α.

It remains to show how to compute f. This can actually be done by a kind of circular sweep around p in $O(m)$ time and space if the vertices of G are given in the order in which they appear on G. During the sweep the problem is to find out on the fly which edges do and which do not contribute to f.

However, since a runtime of $O(m \log m)$ suffices for the sequel, we can use a much more general approach, namely again the computation of lower envelopes. Let ℓ_i be the line that goes through the polygon vertices v_i and $v_{i+1 \bmod m}$ and let β_i be the angle that ℓ_i (directed upwards) forms with $\overrightarrow{h_p}$, see Figure 4. Let f_i be the function that maps $0 \leq \alpha < 2\pi$ to the radius of the unique circle that touches ℓ_i and p and makes the ray from p to its center $c_i(\alpha)$ form an angle of α with $\overrightarrow{h_p}$. f_i has a unique minimum at $\beta_i + \pi/2$ (to which it is symmetric modulo 2π) and a discontinuity at $\beta_i + 3\pi/2$.

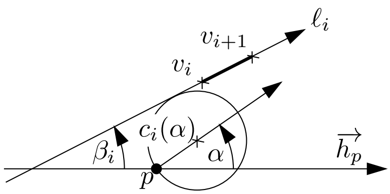

Fig. 4. Notation.

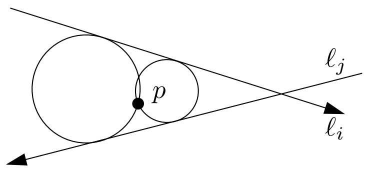

Fig. 5. There are exactly two circles that touch ℓ_i, ℓ_j and p.

Two functions f_i and f_j with $i \neq j$ intersect exactly twice, namely in the angles that correspond to the two circles that touch ℓ_i, ℓ_j and p, see Figure 5. (Note that if ℓ_i and ℓ_j are parallel, p has to lie between them.) The f_i can be represented (and these representations can be computed in constant time each) such that the intersections of f_i and f_j can be computed in constant time for each pair (i, j). Thus we can use the simple devide-and-conquer algorithm described in [AS99][Theorem 2.8] to compute the lower envelope of $f_1, \ldots, f_m$ in $O(m \log m)$ time using $O(m)$ space. The lower envelope of these functions corresponds to f, since the minimum over $f_1, \ldots, f_m$ gives us, for each angle α, the radius of the largest (open) circle that does not intersect any of the constraining lines $\ell_1, \ldots, \ell_m$ and whose center lies in the desired direction relative to p. The discontinuities of the f_i do not matter; they disappear already after the first merge operation of the devide-and-conquer algorithm. ♧

Theorem 1. *A set P of n points in the plane can be labeled with $2n$ non-intersecting circular labels, two per point, of diameter $2/3$ times the optimal label size in $O(n \log n)$ time using linear space.*

Proof. Our algorithm is as follows. First we compute the Voronoi diagram of P. This takes $O(n \log n)$ time and linear space [Aur91]. Then for each input point p we use the algorithm described in the proof of Lemma 4 to compute the largest pair of circles that labels p within the Voronoi cell $\text{Vor}(p)$ of p. Let d_p be the diameter of these circles. Let $m_p < n$ be the number of edges of $\text{Vor}(p)$. Since the complexity of the Voronoi diagram is linear [Aur91], we have $\sum_{p \in P} m_i =$

$O(n)$. Thus, with Lemma 4, we can place all labels in $O(\sum_{p \in P} m_i \log m_i) = O(\sum_{p \in P} m_i \log n) = O(n \log n)$ time in total, using $O(\sum_{p \in P} m_i) = O(n)$ space. We set $d_{\mathrm{algo}} = \min_{p \in P} d_p$ and go through all input points once more. We scale the labels of each point p by a factor of $\frac{d_{\mathrm{algo}}}{d_p}$ using p as the scaling center. We output these scaled labels, all of which now have diameter d_{algo}. Clearly this algorithm runs in $O(n \log n)$ time and uses linear space.

Its correctness can be seen as follows. Since $\frac{d_{\mathrm{algo}}}{d_p} \leq 1$, the scaled labels lie completely in the original labels. Each of these lies in its label zone which in turn lies in the corresponding Voronoi cell according to Lemma 2. Thus no two of the scaled labels intersect. It remains to show that $d_{\mathrm{algo}} \geq d_{\mathrm{lower}}$.

Lemma 1 guarantees that for each input point p there is an orientation (namely that determined by the centers of the labels of p in $\mathcal{C}_{\mathrm{opt}}$) such that the label zone $Z_{\mathrm{label}}(p)$ lies completely within $\mathrm{Vor}(p)$. We do not know this orientation, but Lemma 3 asserts that there are two disjoint labels for p, both of diameter d_{lower}, that lie within $Z_{\mathrm{label}}(p)$—and thus within $\mathrm{Vor}(p)$. This is true even for a point q that receives the smallest labels before scaling, i.e. $d_q = d_{\mathrm{algo}}$. On the other hand we have $d_q \geq d_{\mathrm{lower}}$ since d_q is the maximum label diameter for a pair of labels for q that lies completely within $\mathrm{Vor}(q)$. ♂

Acknowledgments. We are extremely indebted to Tycho Strijk, Utrecht University, for pointing out a mistake in the proof of Lemma 2.

References

[AF84] John Ahn and Herbert Freeman. AUTONAP - an expert system for automatic map name placement. In *Proceedings International Symposium on Spatial Data Handling*, pages 544–569, 1984.

[AS99] Pankaj Agarwal and Micha Sharir. Davenport-Schinzel sequences and their geometric applications. In Jörg-Rüdiger Sack and Jorge Urrutia, editors, *Handbook of Computational Geometry*. Elsevier Science Publishers B.V. North-Holland, Amsterdam, 1999.

[Aur91] Franz Aurenhammer. Voronoi diagrams: A survey of a fundamental geometric data stru cture. *ACM Comput. Surv.*, 23(3):345–405, September 1991.

[C+96] Bernard Chazelle et al. Application challenges to computational geometry: CG impact task force report. Technical Report TR-521-96, Princeton University, April 1996. `http://www.cs.princeton.edu/~chazelle/taskforce/CGreport.ps`.

[CMS95] Jon Christensen, Joe Marks, and Stuart Shieber. An empirical study of algorithms for point-feature label placement. *ACM Transactions on Graphics*, 14(3):203–232, 1995.

[DF89] Jeffrey S. Doerschler and Herbert Freeman. An expert system for dense-map name placement. In *Proc. Auto-Carto 9*, pages 215–224, 1989.

[DMM+97] Srinivas Doddi, Madhav V. Marathe, Andy Mirzaian, Bernard M.E. Moret, and Binhai Zhu. Map labeling and its generalizations. In *Proceedings of the 8th ACM-SIAM Symposium on Discrete Algorithms (SODA'97)*, pages 148–157, New Orleans, LA, 4–7 January 1997.

[DMM00] Srinivas Doddi, Madhav V. Marathe, and Bernard M.E. Moret. Point labeling with specified positions. In *Proc. 16th Annu. ACM Sympos. Comput. Geom. (SoCG'00)*, pages 182–190, Hongkong, 12–14 June 2000.

[FW91] Michael Formann and Frank Wagner. A packing problem with applications to lettering of maps. In *Proc. 7th Annu. ACM Sympos. Comput. Geom. (SoCG'91)*, pages 281–288, 1991.

[Hir82] Stephen A. Hirsch. An algorithm for automatic name placement around point data. *The American Cartographer*, 9(1):5–17, 1982.

[KKSY99] Sung Kwon-Kim, Chan-Su Shin, and Tae-Cheon Yang. Placing two disks in a convex polygon. Technical Report HKUST-TCSC-99-07, Hong Kong University of Science and Technology, Theoretical Computer Science Center, 1999.

[KSY99] Sung Kwon Kim, Chan-Su Shin, and Tae-Cheon Yang. Labeling a rectilinear map with sliding labels. Technical Report HKUST-TCSC-1999-06, Hongkong University of Science and Technology, July 1999.

[KT98] Konstantinos G. Kakoulis and Ioannis G. Tollis. On the multiple label placement problem. In *Proc. 10th Canadian Conf. Computational Geometry (CCCG'98)*, pages 66–67, 1998.

[Mor80] Joel L. Morrison. Computer technology and cartographic change. In D.R.F. Taylor, editor, *The Computer in Contemporary Cartography*. J. Hopkins Univ. Press, New York, 1980.

[QWXZ00] Zhongping Qin, Alexander Wolff, Yinfeng Xu, and Binhai Zhu. New algorithms for two-label point labeling. In *Proc. 8th Annu. Europ. Symp. on Algorithms (ESA'00)*, Lecture Notes in Computer Science, Saarbrücken, 5–8 September 2000. Springer-Verlag. to appear.

[SW00] Tycho Strijk and Alexander Wolff. Labeling points with circles. *International Journal of Computational Geometry and Applications*, 2000. to appear.

[vKSW99] Marc van Kreveld, Tycho Strijk, and Alexander Wolff. Point labeling with sliding labels. *Computational Geometry: Theory and Applications*, 13:21–47, 1999.

[WS96] Alexander Wolff and Tycho Strijk. A map labeling bibliography. `http://www.math-inf.uni-greifswald.de/map-labeling/bibliography/`, 1996.

[WW97] Frank Wagner and Alexander Wolff. A practical map labeling algorithm. *Computational Geometry: Theory and Applications*, 7:387–404, 1997.

[Zor90] Steven Zoraster. The solution of large 0-1 integer programming problems encountered in automated cartography. *Operations Research*, 38(5):752–759, 1990.

[ZP99] Binhai Zhu and Chung Keung Poon. Efficient approximation algorithms for multi-label map labeling. In A. Aggarwal and C. Pandu Rangan, editors, *Proc. 10th Annual International Symposium on Algorithms and Computation (ISAAC'99)*, *Lecture Notes in Computer Science* 1741, pages 143–152, Chennai, India, 16–18 December 1999. Springer-Verlag.

[ZQ00] Binhai Zhu and Zhongping Qin. New approximation algorithms for map labeling with sliding labels. Dept. of Computer Science, City University of Hong Kong, 2000.