# Polynomial Time Algorithms for Label Size Maximization on Rotating Maps

Yusuke Yokosuka[*] Keiko Imai[†]

## Abstract

Map labeling is a problem of placing labels at corresponding graphical features on a map. There are two optimization problems: the label number maximization problem and the label size maximization problem. In general, both problems are NP-hard for static maps. Recently, the widespread use of several applications, such as personal mapping systems, has increased the importance of dynamic maps and the label number maximization problem for dynamic cases has been studied. In this paper, we consider the label size maximization problem for points on rotating maps. Our model is as follows. For each label, a point is chosen inside the label or on its boundary as an anchor point. Each label is placed such that the anchor point coincides with the corresponding point on the map. Furthermore, while the map fully rotates from 0 to $2\pi$, the labels are placed horizontally according to the angle of the map. Our problem consists of finding the maximum scale factor for the labels such that the labels do not intersect, and deciding the place of the anchor points. We propose an $O(n \log n)$-time and $O(n)$-space algorithm for the case where each anchor point is inside the label. Moreover, if the labels are of unit-height (or unit-width) and the anchor points are on the boundary, we also present an $O(n \log n)$-time and $O(n)$-space algorithm.

## 1 Introduction

*Map labeling* is the problem of placing text or symbol labels corresponding to graphical features on input maps such that the labels are pairwise disjoint. This problem is important in several areas, such as geographic information system (GIS), cartography, and graph drawing. On maps, labels of regions, rivers, stations, etc., are placed in appropriate positions so that the corresponding features in the map can be understood. In map labeling, points, polylines, and polygons are considered as graphical features. In this paper, we consider map labeling for points.

A lot of map labeling research has been presented [15]. There are two optimization problems in map labeling. One is the *label number maximization problem* of finding the placement of a maximum cardinality subset of labels with fixed size. The other is the *label size maximization problem* of placing all labels such that the sizes of the labels are maximized under a global scale factor. Most research has considered static maps.

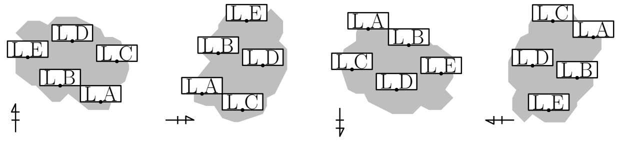


Figure 1: Example of label size maximization problem for rotating maps.

Recently, the importance of dynamic maps has increased due to several applications such as personal mapping systems. There are a lot of dynamic cases, for example, panning, rotating, and zooming maps, translating points, moving points with different velocity. In this context, research on map labeling for dynamic cases has been presented [2, 3, 9, 10]. Mainly, the dynamic label number maximization problem was investigated in their research. In contrast to this, it is a natural direction to consider label size maximization problems for dynamic maps.

In this paper, we consider rotating maps. Since commercial GIS applications (e.g., navigation) often rotate maps dynamically according to the direction in which the user is facing, we assume that labels are placed horizontally according to the angle of the map. We consider the problem of maximizing the label size such that the labels are pairwise disjoint over all rotations $\theta \in [0, 2\pi)$ (Figure 1).

### 1.1 Problem Definition and Our Results

Let $M$ be a map that includes a set of points $P = \{p_1, \ldots, p_n\}$ in the plane with a set of labels $L = \{\ell_1, \ldots, \ell_n\}$. In this paper, the labels are considered to be open axis-aligned rectangles of different sizes. Each initial size of $\ell_i \in L$ is expressed by its width $w_i > 0$ and height $h_i > 0$. When the *scale factor* is $\sigma$, the label size of $\ell_i$ is $w_i\sigma \times h_i\sigma$.

Each label is placed such that a point called an *anchor point* coincides with the corresponding point $p_i$ (Figure 2 (a)). The anchor point is inside the label $\ell_i$ or on its boundary. When the label $\ell_i$ is fixed in place,

[*] Department of Information and System Engineering, Graduate School of Science and Engineering, Chuo University, yoyu@imai-lab2.ise.chuo-u.ac.jp

[†] Department of Information and System Engineering, Chuo University, imai@ise.chuo-u.ac.jp

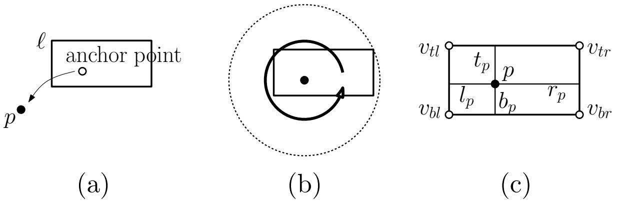


Figure 2: Definitions.

Table 1: Our results (running time). All solvable problems can be computed in $O(n)$ space.

| Rectangle shape | MSR | MSBR |
|---|---|---|
| Unit squares | $O(n \log n)$<br>(Theorem 4) | $O(n \log n)$<br>(Corollary 5) |
| Squares | | - |
| Unit-height rectangles | $O(n \log n)$<br>(Theorem 6) | $O(n \log n)$<br>(Corollary 9) |
| Rectangles | | - |

we say that it is *anchored* at $p_i$. While $M$ fully rotates from 0 to $2\pi$ with the anchor points touching the corresponding points, the labels are placed horizontally, and should not intersect each other. Our problem is finding the *maximum scale factor* $\sigma^*$ such that the labels do not intersect, and deciding the place of the anchor points. In ordinary map labeling, each label is placed such that the anchor point is on the boundary of its label. However, we also consider the case that the point is inside the label. We call the former problem the *maximization problem of the size of labels with boundary anchor points on rotating maps* (*MSBR*), and the latter problem the *maximization problem of the size of labels on rotating maps* (*MSR*). This formulation on dynamic maps is a natural extension of the label size maximization problem on static maps.

Our results are summarized in Table 1. We address several rectangular label shapes (e.g., unit squares and unit-height rectangles). Although static label size maximization is NP-hard [8], MSR and MSBR can be solved in polynomial time, which is surprising.

In the following, we treat the clockwise rotation of $M$ as the counterclockwise rotation of labels around their anchor points (Figure 2 (b)), as did Gemsa et al. [9]. Both rotations are equivalent and yield exactly the same results.

### 1.2 Related Work

In map labeling, two models have been considered w.r.t. the number of label candidates for each point: the *fixed-position model* [8] and the *slider model* [14]. In both models, each label is placed such that the corresponding point is on the boundary of the label. The fixed-position model has a finite number of label candidates (e.g., the 2-position and 4-position model). The label candidates of the slider model are the specified sides of the labels (e.g., in the 2-slider model, two sides of the label serve as a set of label candidates).

It is known that the static label size maximization problems, except for the 1-position and 2-position model, are APX-hard, even for unit square labels [8]. A lot of constant-factor approximation algorithms have been proposed for several axis-parallel rectangles [8, 11]. Doddi et al. [6] dealt with unit square labels with different orientations, and Zhu and Qin [16] considered the case that all the square labels have the same orientation. Furthermore, the static label number maximization problems in several models are known to be NP-hard (e.g., [8, 14]). Therefore, many approximation algorithms have already been presented (e.g., [1, 14]).

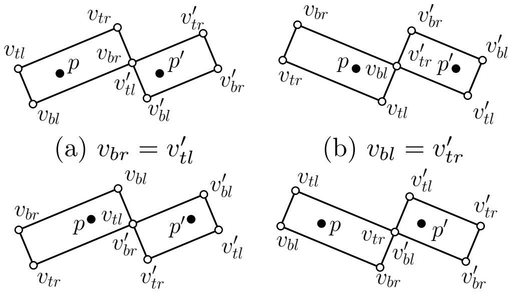
In dynamic map labeling, Been et al. [2] proposed consistency desiderata for dynamic map labeling, which are that labels should not pop and jump during panning and zooming. Been et al. [3] treated the problems of maximizing the lengths of active ranges, where the active range of a label $\ell$ is a contiguous range of map scales at which $\ell$ is displayed. Moreover, the problem satisfies that the labels are pairwise disjoint at any scale and satisfy the consistency desiderata. They proved that the problems for points in the plane are NP-hard, and proposed several exact and approximation algorithms for points in 1D and 2D. Gemsa et al. [10] extended the above problems to the slider model, and also dealt with selecting the slider positions. Moreover, Gemsa et al. [9] considered similar dynamic map labeling for rotating maps. They also proved that the problem is NP-hard, and proposed approximation algorithms.

In the circular labeling problem [13], the corresponding point in the plane is on the boundary of the circular label. However, in MSR and MSBR, during the rotation, the point is inside the label or on the boundary, and it may not be on the circle obtained by rotation of the label.

## 2 Properties

In this section, first, we investigate locations of anchor points such that the scale factor is maximized. Next, for the locations, we calculate the maximum scale factor.

Let $\ell_p$ be a label anchored at a point $p$ with the initial width $w_p$ and the initial height $h_p$. Further, the top-left, top-right, bottom-left, and bottom-right point of $\ell$ rotated by angle 0 are denoted by $v_{tl}$, $v_{tr}$, $v_{bl}$, and $v_{br}$, respectively. Draw the segments passing through $p$ in parallel with the edges of $\ell_p$. We assume that $p$ divides the horizontal segment and vertical segment internally in the ratio $l_p : r_p$ (where $r_p = 1 - l_p$) and $t_p : b_p$ (where $b_p = 1 - t_p$), respectively (Figure 2 (c)). We define each parameter for a point $p'$ in the same way. Thus, $\ell_{p'}$ is



(a) $v_{br} = v'_{tl}$ (b) $v_{bl} = v'_{tr}$

(c) $v_{tl} = v'_{br}$ (d) $v_{tr} = v'_{bl}$

Figure 3: Four possible cases that the corner points of two labels $\ell_p$ and $\ell_{p'}$ intersect.

the label of $p'$. $w_{p'}$ and $h_{p'}$ are the initial width and initial height of $\ell_{p'}$, respectively. Moreover, the above parameters of $\ell_{p'}$ are defined as $v'_{tl}$, $v'_{tr}$, $v'_{bl}$, $v'_{br}$, $l_{p'}$, $r_{p'}$, $t_{p'}$ and $b_{p'}$. Finally, let $d$ be the distance between $p$ and $p'$, and $\sigma_{pp'}$ be the maximum scale factor for $p$ and $p'$.

**Lemma 1** *Let $\ell_p$ and $\ell_{p'}$ be two labels, which are anchored at points $p$ and $p'$, respectively. $\ell_p$ and $\ell_{p'}$ can be placed with the maximum scale factor $\sigma_{pp'}$ if and only if the anchor points of $\ell_p$ and $\ell_{p'}$ satisfy $(1-2l_p)w_p = (1-2l_{p'})w_{p'}$ and $(1-2t_p)h_p = (1-2t_{p'})h_{p'}$.*

**Proof.** Without loss of generality, we assume that $p$ and $p'$ lie on a horizontal line. Let $\sigma$ be the scale factor for $p$ and $p'$. Note that $\ell_p$ and $\ell_{p'}$ touch at their corner points. Otherwise, if $\ell_p$ and $\ell_{p'}$ touch on their boundary segments, they overlap by slight rotation. Moreover, $\ell_p$ and $\ell_{p'}$ are parallel. Therefore, there are only the following four possible cases: $v_{br} = v'_{tl}$, $v_{bl} = v'_{tr}$, $v_{tl} = v'_{br}$, and $v_{tr} = v'_{bl}$ (Figure 3).

We consider the case that $v_{tl} = v'_{br}$ (Figure 4). Since $\ell_p$ and $\ell_{p'}$ are parallel, we have $(l_p w_p \sigma + (1-l_{p'})w_{p'}\sigma)^2 + (t_p h_p \sigma + (1-t_{p'})h_{p'}\sigma)^2 \leq d^2$. Therefore,

$$\sigma \leq d/\sqrt{(l_p w_p + (1-l_{p'})w_{p'})^2 + (t_p h_p + (1-t_{p'})h_{p'})^2}. \quad (1)$$

In the same way, if $v_{tr} = v'_{bl}$,

$$\sigma \leq d/\sqrt{((1-l_p)w_p + l_{p'}w_{p'})^2 + (t_p h_p + (1-t_{p'})h_{p'})^2}, \quad (2)$$

if $v_{br} = v'_{tl}$,

$$\sigma \leq d/\sqrt{((1-l_p)w_p + l_{p'}w_{p'})^2 + ((1-t_p)h_p + t_{p'}h_{p'})^2}, \quad (3)$$

and if $v_{bl} = v'_{tr}$,

$$\sigma \leq d/\sqrt{(l_p w_p + (1-l_{p'})w_{p'})^2 + ((1-t_p)h_p + t_{p'}h_{p'})^2}. \quad (4)$$

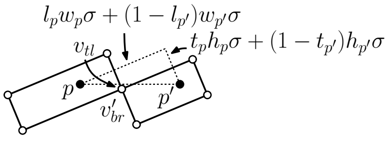

$l_p w_p \sigma + (1-l_{p'})w_{p'}\sigma$
$v_{tl}$
$t_p h_p \sigma + (1-t_{p'})h_{p'}\sigma$
$p$
$v'_{br}$
$p'$

Figure 4: The case that $v_{tl} = v'_{br}$.

First, we focus on the inequalities (1) and (2) (or, (3) and (4)). As the denominators of the right-hand sides of (1) and (2) become smaller, the maximum possible $\sigma$ becomes greater. $(t_p h_p + (1-t_{p'})h_{p'})^2$ is appeared in both right-hand sides of (1) and (2). The smaller $(l_p w_p + (1-l_{p'})w_{p'})^2$ becomes, the greater $((1-l_p)w_p + l_{p'}w_{p'})^2$ becomes. Therefore, if $(l_p w_p + (1-l_{p'})w_{p'})^2 = ((1-l_p)w_p + l_{p'}w_{p'})^2$, $\sigma$ is maximized among values satisfying (1) and (2). This condition is equivalent to the equation $(1-2l_p)w_p = (1-2l_{p'})w_{p'}$. Similarly, in case that the inequalities (1) and (4) (or, (2) and (3)), if $(1-2t_p)h_p = (1-2t_{p'})h_{p'}$, $\sigma$ is maximized among values satisfying (1) and (4). The above two equations are satisfied simultaneously. Therefore, if $(1-2l_p)w_p = (1-2l_{p'})w_{p'}$ and $(1-2t_p)h_p = (1-2t_{p'})h_{p'}$, $\sigma$ is the maximum scale factor $\sigma_{pp'}$.

The converse is also true. □

From Lemma 1, we can obtain the following lemma.

**Lemma 2** *For given points $p$ and $p'$ with labels $\ell_p$ and $\ell_{p'}$, if the anchor point of each label is the intersection of two diagonals of the label, $\ell_p$ and $\ell_{p'}$ can be placed with the maximum scale factor $\sigma_{pp'}$.*

**Proof.** Let $p$ and $p'$ be two points. In the case that the anchor points lie in the label centers, $l_p = t_p = l_{p'} = t_{p'} = 1/2$. Therefore, $(1-2\times 1/2)w_p = (1-2\times 1/2)w_{p'} = 0$ and $(1-2\times 1/2)h_p = (1-2\times 1/2)h_{p'} = 0$. The conditions in Lemma 1 are satisfied, and hence, the scale factor $\sigma_{pp'}$ is maximized. □

From Lemma 2, the maximum scale factor $\sigma_{pp'}$ of MSR for $p$ and $p'$ is $2d/\sqrt{(w_p + w_{p'})^2 + (h_p + h_{p'})^2}$. Therefore, we can solve MSR for more than two points by computing the maximum scale factor $\sigma_{ij}$ for all point pairs $p_i$ and $p_j$, and choosing the minimum among those. This naive algorithm runs in $\Theta(n^2)$ time. Moreover, if all heights (or widths) of labels are equal to each other, we obtain the following proposition.

**Proposition 3** *MSBR for unit-height (or unit-width) rectangular labels can be computed in $\Theta(n^2)$ time.*

**Proof.** The naive algorithm of MSR gives the maximum scale factor $\sigma^*$ for the unit-height rectangular labels. We consider points obtained by translating the anchor points placed at the center of rectangles in MSR to the top or bottom (or, left or right) boundary. Those points satisfy the equations (1)–(4) in Lemma 1. Therefore, the points are the anchor points in MSBR and $\sigma^*$ is also the maximum scale factor in MSBR. □

In the following sections, we will improve the time complexity of these algorithms for MSR and MSBR to $O(n \log n)$.

## 3 Square Labels

When the labels of all points are squares, the problem has a strong connection to the *weighted closest pair problem* [7]: The input is a set of disks. Each disk has a point in $P$ as its center, a weight $W$, and a radius $W\sigma$, where $\sigma$ is a scale factor. The goal is to find the maximum scale factor $\sigma^*$ such that the disks are pairwise disjoint.

**Theorem 4** *MSR for square labels can be computed in $O(n \log n)$ time and $O(n)$ space.*

**Proof.** Let $p$ and $p'$ be two points that have square labels, and lie on a horizontal line. Let $\sigma_{pp'}$ be the maximum scale factor for $p$ and $p'$. Since the labels are square, we have $w_p = h_p$ and $w_{p'} = h_{p'}$. When the labels are anchored at $p$ and $p'$, and their anchor points are their centers by Lemma 2, the distance between $p$ and $p'$ is $\frac{\sqrt{2}}{2}(w + w')\sigma_{pp'}$. Then, $\sigma_{pp'}$ is determined by the angles $\pi/4$, $3\pi/4$, $5\pi/4$, and $7\pi/4$. We consider the disks drawn by fully rotating the square labels around the points $p$ and $p'$. The maximum scale factor $\sigma_{pp'}$ is obtained by maximizing the size of the disks such that they are pairwise disjoint.

Therefore, MSR for square labels is considered as the weighted closest pair problem with weight $W = \frac{\sqrt{2}}{2} w_p$ for each point $p$. For the weighted closest pair problem, Formann [7] proposed an $O(n \log n)$-time and $O(n)$-space algorithm based on a plane sweep. Therefore, this completes the proof. $\square$

**Corollary 5** *MSBR for unit square labels can be computed in $O(n \log n)$ time and $O(n)$ space.*

## 4 Rectangular Labels

For the rectangular labels, the algorithm of square labels does not work directly because the disks obtained by sweeping the rectangular labels around their anchor points can intersect when the scale factor is maximized. However, Formann's idea [7] used in weighted closest pair problem can be modified to MSR and MSBR for rectangular labels. Our modified algorithm overestimates the maximum scale factor, and then fixes the maximum value using the intersection graph of disks drawn by fully rotation of the labels. In the algorithm, we use the *Delaunay triangulation* [5, 12] of $P$, $\mathrm{DT}(P)$, which is a triangulation with the *empty circle property*: for any triangle $T$ in $\mathrm{DT}(P)$, the circumcircle of $T$ contains no points of $P$ in its interior. We call a triangle of $\mathrm{DT}(P)$ a *Delaunay triangle*. When points $p$ and $q$ are vertices of a Delaunay triangle in $\mathrm{DT}(P)$, $q$ is called a *neighbor* of $p$.

Our algorithm can be described as Algorithm 1.

**Algorithm 1** Algorithm for MSR.

```
1: Compute $\mathrm{DT}(P)$ for $P$.
2: For each point $p$, calculate the maximum scale factor $\sigma_p$ with all the neighbors in $\mathrm{DT}(P)$. Take the minimum scale factor $\sigma_{\mathrm{pre}} = \min_{p \in P} \sigma_p$ of all the scale factors.
3: For each point $p \in P$, draw a closed disk with center $p$ and radius $\frac{\sigma_{\mathrm{pre}}}{2}\sqrt{w_p^2 + h_p^2}$. Enumerate all intersections of disks using the standard intersection detecting algorithm of Bentley and Ottmann [4].
4: Calculate the maximum scale factor for all intersections of disks, and take the minimum value among them as $\sigma^*$.
```

The following theorem shows the correctness of Algorithm 1 and its complexity. In the following, let $D_p$ be the disk centered at $p$ in Step 3 of Algorithm 1, and let $R_p$ be its radius $\frac{\sigma_{\mathrm{pre}}}{2}\sqrt{w_p^2 + h_p^2}$.

**Theorem 6** *MSR can be computed in $O(n \log n)$ time and $O(n)$ space.*

In order to prove Theorem 6, we present some lemmas.

**Lemma 7** *Each disk obtained after Step 3 of Algorithm 1 contains no points in $P$ other than its center point.*

**Proof.** For each $p \in P$, let $D_p$ be the disk with center $p$ and radius $R_p = \frac{\sigma_{\mathrm{pre}}}{2}\sqrt{w_p^2 + h_p^2}$. From the definition of $\sigma_{\mathrm{pre}}$, the labels of $p$ and $q$ do not intersect during rotation for a neighbor $q$. Therefore, $D_p$ cannot contain neighbors of $p$. In the Delaunay triangulation, the nearest point $q$ of $p$ is a neighbor of $p$ in $\mathrm{DT}(P)$. Therefore, the radius of $D_p$ is less than $|\overline{pq}|$. From this, $D_p$ cannot contain points that are not the neighbors of $p$ in $DT(P)$. $\square$

**Lemma 8** *The number of intersecting pairs in the set of disks obtained at Step 3 of Algorithm 1 is at most $3n - 6$.*

**Proof.** First, we draw straight line segments between the points whose closed disks intersect at Step 3 of Algorithm 1. We will show that the straight line graph $G$ having the line segments as edges is planar. We consider the case that two closed disks $D_p$ and $D_{p'}$ intersect. In $G$, $p$ and $p'$ are connected by a straight line edge. If there is no other disk $D_q$ centered at a point $q \neq p, p'$

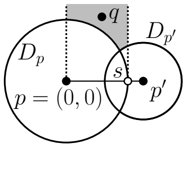


Figure 5: Assumption of Lemma 8.

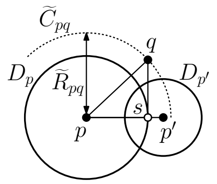


Figure 6: Case 1.

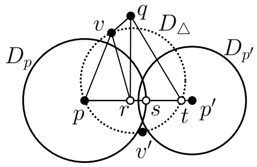


Figure 7: Case 2-1.

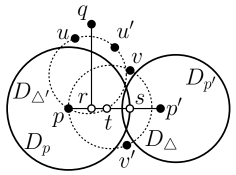


Figure 8: Case 2-2a.

which intersects the line segment $\overline{pp'}$, no two line segments in $G$ intersect without endpoints. This shows that the graph $G$ is planar.

Without loss of generality, we assume that $p$ and $p'$ lie on a horizontal line and the $x$-coordinate of $p'$ is greater than that of $p$ (Figure 5). We denote the $x$- and $y$-coordinates of $p$ and $p'$ by $x_p$, $y_p$, $x_{p'}$, and $y_{p'}$, respectively. We denote the $x$- and $y$-coordinates of the other points in the same way. Let $s$ be the intersection of the boundary of $D_p$ and $\overline{pp'}$. In the following, we assume $0 = x_p \le x_q \le x_s$ and $0 = y_p = y_{p'} \le y_q$. $q$ is in the shaded area in Figure 5. When $x_q < 0$ or $x_{p'} < x_q$, $D_p$ cannot intersect $\overline{pp'}$, by Lemma 7. Moreover, when $y_q < 0 = y_p = y_{p'}$ or $x_s < x_q \le x_{p'}$, these cases can be proved in the same way.

We consider separately the cases that $q$ is or is not a neighbor of $p$ in $\mathrm{DT}(P)$.

**Case 1:** $q$ is a neighbor of $p$ in $\mathrm{DT}(P)$.

We denote $\frac{\sigma_{\mathrm{pre}}}{2}\sqrt{(w_p+w_q)^2+(h_p+h_q)^2}$ by $\widetilde{R}_{pq}$. In this case, by the definition of $\sigma_{\mathrm{pre}}$, we have $|\overline{pq}| \ge \widetilde{R}_{pq}$. Moreover, since $w_q, h_q > 0$, $\widetilde{R}_{pq}$ is greater than $R_p$. Let $\widetilde{C}_{pq}$ be a circle centered at $p$ with radius $\widetilde{R}_{pq}$. $\widetilde{C}_{pq}$ is shown as a dotted circle in Figure 6. Note that the vertical distance between $q$ and $\overline{pp'}$ is greater than or equal to the length of a vertical straight segment from $s$ to $\widetilde{C}_{pq}$. Then, we consider the case that $x_q = x_s$. Because $|\overline{pq}| \ge \widetilde{R}_{pq}$ and $|\overline{ps}| = R_p = \frac{\sigma_{\mathrm{pre}}}{2}\sqrt{w_p^2+h_p^2}$, we have that

$$\begin{aligned}
|\overline{sq}|^2 &= |\overline{pq}|^2 - |\overline{ps}|^2 \\
&\ge \left(\frac{\sigma_{\mathrm{pre}}}{2}\right)^2((w_p+w_q)^2+(h_p+h_q)^2) \\
&\quad -\left(\frac{\sigma_{\mathrm{pre}}}{2}\right)^2(w_p^2+h_p^2) \\
&= \left(\frac{\sigma_{\mathrm{pre}}}{2}\right)^2(w_q^2+h_q^2+2w_pw_q+2h_ph_q).
\end{aligned}$$

Since $R_q = \frac{\sigma_{\mathrm{pre}}}{2}\sqrt{w_q^2+h_q^2}$ and $w_p, h_p, w_q, h_q > 0$, we have that $|\overline{sq}|^2 - R_q^2 \ge \left(\frac{\sigma_{\mathrm{pre}}}{2}\right)^2(2w_pw_q+2h_ph_q) > 0$. Therefore, $D_q$ cannot intersect $\overline{pp'}$.

**Case 2:** $q$ is not a neighbor of $p$ in $\mathrm{DT}(P)$.

In this case, we can show that there is a Delaunay triangle with $p$ whose circumcircle contains $\overline{pp'} \cap D_p$ in the following way. If $p'$ is a neighbor of $p$, the circumcircle of a Delaunay triangle that has the edge $\overline{pp'}$ completely contains $\overline{pp'} \cap D_p$. If $p'$ is not a neighbor of $p$, there is a Delaunay triangle $\triangle pvv'$ that has an edge that intersects the inside of $\overline{pp'}$. Since $v, v' \notin D_p$ by Lemma 7, the circumcircle of $\triangle pvv'$ completely contains $\overline{pp'} \cap D_p$. Therefore, we consider such a Delaunay triangle $\triangle pvv'$. In this case, $v'$ might be $p'$. Let $v$ and $v'$ be points such that $y_v > 0$ and $y_{v'} \le 0$, respectively. Further, let $t$ be the intersection point of the circumcircle of $\triangle pvv'$ and $\overline{pp'}$. Since the circumcircle contains $\overline{pp'} \cap D_p$ regardless of whether $p'$ is a neighbor of $p$, we have $x_q \le x_s < x_t$. In the case that $x_v = x_q$, since $y_v < y_q$ and by Lemma 7, $D_q$ cannot intersect $\overline{pp'}$. Therefore, we consider two cases: $x_v < x_q \le x_s$ and $x_q < x_v < x_s$. In the following, we denote the closed disk whose boundary is the circumcircle of $\triangle pvv'$ by $D_\triangle$.

**Case 2-1:** $x_v < x_q \le x_s$.

Let $t$ be the intersection of $\overline{pp'}$ and the boundary of $D_\triangle$, and $r$ be the intersection of $\overline{pp'}$ and a perpendicular from $q$ to $\overline{pp'}$ (Figure 7). Since $D_\triangle$ completely contains $\overline{pp'} \cap D_p$, $x_r \le x_s < x_t$. Because $\angle prq = \pi/2$ and $x_t - x_s > 0$, $\angle ptq < \pi/2$. Moreover, since $q \notin D_\triangle$ by the empty circle property, $\angle pvq + \angle ptq \ge \pi$. Therefore, we have $\angle pvq \ge \pi - \angle ptq > \pi/2$. Because $r \in D_p$ and $v \notin D_p$, $|\overline{pr}| < |\overline{pv}|$ and $\angle pvr \le \angle prv$. Then, we have $\angle qvr = \angle pvq - \angle pvr > \pi/2 - \angle prv = \angle qrv$. Therefore, we have $|\overline{qv}| < |\overline{qr}|$. Since $R_q < |\overline{qv}|$ by Lemma 7, $D_q$ cannot contain $r$. Therefore, $D_q$ cannot intersect $\overline{pp'}$.

**Case 2-2:** $x_q < x_v < x_s$.

First, we show that there is a Delaunay triangle $\triangle puu'$ such that $x_u \le x_q \le x_{u'}$. Since $q$ is not a neighbor of $p$ in $\mathrm{DT}(P)$, there is a Delaunay triangle $\triangle pzz'$ that has an edge $\overline{zz'}$ that intersects $\overline{pq}$ at an interior point of $\overline{pq}$. Moreover, $v$ is a vertex of a Delaunay triangle that has $p$ as its vertex. Since $p$ is in the polygon consisting of Delaunay triangles that have $p$ as their vertex, when we visit the Delaunay triangles clockwise from $\triangle pzz'$ to $\triangle pvv'$, there is a Delaunay triangle $\triangle puu'$ that satisfies the condition. In the following, we denote the closed disk whose boundary is the circumcircle of $\triangle puu'$ by $D_{\triangle'}$, and the boundary of $D_{\triangle'}$ by $C_{\triangle'}$. We consider the cases that $r \in D_{\triangle'} \setminus C_{\triangle'}$ and $r \notin D_{\triangle'} \setminus C_{\triangle'}$.

**Case 2-2a:** $r \in D_{\triangle'} \setminus C_{\triangle'}$.

Let $t$ be the intersection of $C_{\triangle'}$ and the inside of $\overline{pp'}$ (Figure 8). Since $r \in D_{\triangle'} \setminus C_{\triangle'}$, we have $x_r < x_t$. Therefore, we can use the same proof as for Case 2-1 by

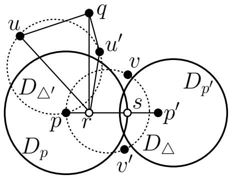


Figure 9: Case 2-2b.

replacing $u$ with $v$. It is shown that $D_q$ cannot intersect $\overline{pp'}$.

**Case 2-2b:** $r \notin D_{\triangle'} \setminus C_{\triangle'}$.

In this case, we consider the quadrilateral $uqu'r$ (Figure 9). By Lemma 7, $u$ is not contained inside $D_q$. Therefore, if $D_q$ intersects $\overline{pp'}$, $|\overline{qr}| \leq R_q < |\overline{qu}|$ or $|\overline{qr}| \leq R_q < |\overline{qu'}|$. First, we consider the case $|\overline{qr}| \leq R_q < |\overline{qu}|$. Since $\angle qur < \angle qru$ in $\triangle qur$, we have $\angle qur < \pi/2$. Since $q$ and $r$ are outside $D_{\triangle'}$, we have $\angle qu'r + \angle qur \geq \pi$. Therefore, we have $\angle qu'r > \pi/2$ and $\angle qru' < \pi/2$. This means that $|\overline{qu'}| < |\overline{qr}| \leq R_q$ and $u'$ is inside $D_q$. This contradicts Lemma 7. The case $|\overline{qr}| \leq R_q < |\overline{qu'}|$ can be proven in the same way. Therefore, $D_q$ cannot intersect $\overline{pp'}$. □

**Proof of Theorem 6.** First, we show the correctness. From the definition of $\sigma_{\text{pre}}$ in Step 2 of Algorithm 1, two labels whose corresponding points are neighbors in $\text{DT}(P)$ do not intersect. In Step 4, the disks can be drawn by fully rotating the labels from 0 to $2\pi$. Each label has the anchor point at its center, and is scaled by $\sigma_{\text{pre}}$. Moreover, since $\sigma_{\text{pre}} \geq \sigma^*$, we can obtain $\sigma^*$ by checking intersecting disks.

Next, we show the complexity. Step 1 can be computed in $O(n \log n)$ time and $O(n)$ space [5, 12]. Step 2 calculates the maximum scale factor between neighbors. Since the number of edges in Delaunay triangulation is $O(n)$, Step 2 can be computed in $O(n)$ time and $O(1)$ space. In Step 3, the algorithm of Bentley and Ottmann [4] can be computed in $O((n+K)\log n)$ time and $O(n+K)$ space where $K$ is the number of intersecting pairs. Moreover, Step 4 can be computed in $O(K)$ time and $O(1)$ space. Since $K \leq 3n-6$ by Lemma 8, this completes the proof. □

**Corollary 9** *MSBR for unit-height (or unit-width) rectangular labels can be computed in $O(n \log n)$ time and $O(n)$ space.*

From Theorem 4, MSR and MSBR are generalizations of the closest pair problem. The time complexity of this problem is lower-bounded by $\Omega(n \log n)$ [12], which may also apply to our problems.

## 5 Conclusion

We considered the label size maximization problem for rotating maps. In general, label size maximization problems for static maps are APX-hard. However, we showed that the problem for rotating maps can be solved in polynomial time, and we presented efficient algorithms for finding the maximum scale factor.

### Acknowledgments.

The work of the second author was supported in part by Grant-in-Aid for Scientific Research of Japan Society for the Promotion of Science.

### References

[1] P. K. Agarwal, M. J. van Kreveld, and S. Suri. Label placement by maximum independent set in rectangles. *Comput. Geom.*, 11(3-4):209–218, 1998.

[2] K. Been, E. Daiches, and C.-K. Yap. Dynamic map labeling. *IEEE Trans. Vis. Comput. Graph.*, 12(5):773–780, 2006.

[3] K. Been, M. Nöllenburg, S.-H. Poon, and A. Wolff. Optimizing active ranges for consistent dynamic map labeling. *Comput. Geom.*, 43(3):312–328, 2010.

[4] J. L. Bentley and T. Ottmann. Algorithms for reporting and counting geometric intersections. *IEEE Trans. Computers*, 28(9):643–647, 1979.

[5] M. de Berg, O. Cheong, M. van Kreveld, and M. Overmars. *Computational Geometry: Algorithms and Applications*. Springer-Verlag, 3rd edition, 2008.

[6] S. Doddi, M. V. Marathe, A. Mirzaian, B. M. E. Moret, and B. Zhu. Map labeling and its generalizations. In *SODA*, pages 148–157, 1997.

[7] M. Formann. Weighted closest pairs. In *STACS*, pages 270–281, 1993.

[8] M. Formann and F. Wagner. A packing problem with applications to lettering of maps. In *ACM Symposium on Computational Geometry*, pages 281–288, 1991.

[9] A. Gemsa, M. Nöllenburg, and I. Rutter. Consistent labeling of rotating maps. In *WADS*, pages 451–462, 2011.

[10] A. Gemsa, M. Nöllenburg, and I. Rutter. Sliding labels for dynamic point labeling. In *CCCG*, pages 205–210, 2011.

[11] J.-W. Jung and K.-Y. Chwa. Labeling points with given rectangles. *Inf. Process. Lett.*, 89(3):115–121, 2004.

[12] F. P. Preparata and M. I. Shamos. *Computational Geometry - An Introduction*. Springer, 1985.

[13] T. Strijk and A. Wolff. Labeling points with circles. *Int. J. Comput. Geometry Appl.*, 11(2):181–195, 2001.

[14] M. J. van Kreveld, T. Strijk, and A. Wolff. Point labeling with sliding labels. *Comput. Geom.*, 13(1):21–47, 1999.

[15] A. Wolff and T. Strijk. The map-labeling bibliography, 2009.

[16] B. Zhu and Z. Qin. New approximation algorithms for map labeling with sliding labels. *J. Comb. Optim.*, 6(1):99–110, 2002.