

Regular Paper

Polynomial Time Algorithms for Label Size Maximization on Rotating Maps

YUSUKE YOKOSUKA^{1,a)} KEIKO IMAI^{2,b)}

Received: November 7, 2016, Accepted: February 9, 2017

Abstract: Map labeling is the problem of placing labels at corresponding graphical features on a map. There are two main 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, an anchor point is chosen inside the label or on its boundary. 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 2π , 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 determining the placing of the anchor points. We describe 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 anchor points are on the boundaries, we also present an $O(n \log n)$ -time and $O(n)$ -space exact and approximation algorithms for several label shapes.

Keywords: map labeling, label size maximization, dynamic maps, rotating maps

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 for several applications, such as geographic information systems (GIS), cartography, and graph drawing. On maps, labels indicating regions, rivers, stations, etc., are placed in appropriate positions so that the corresponding features on 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.

Many studies on map labeling have been presented (see Ref. [23]). There are two main 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 their sizes are maximized under a global scale factor. Most studies were considered static maps.

Recently, the importance of dynamic maps has increased due to several applications such as personal mapping systems. There are many dynamic cases, involving for example, panning, rotating, and zooming maps, translating points, moving points with different velocities. In this context, studies on map label-

ing for dynamic cases have been presented in which mainly the dynamic label number maximization problem has been considered [2], [3], [5], [10], [11], [12], [13], [15], [25]. However, for dynamic maps, there are not many studies for label size maximization problem [16].

In this paper, we consider rotating maps. Since commercial GIS applications (e.g., navigation systems) often rotate maps dynamically according to the direction in which the user is facing, we assume that the 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 $\theta \in [0, 2\pi)$ (Fig. 1).

1.1 Problem Definition and Our Results

Let M be a map that includes a set of points $P = \{p_1, \dots, p_n\}$ in the plane with a set of labels $L = \{\ell_1, \dots, \ell_n\}$. In this paper, the labels are considered to be open axis-aligned rectangles of different sizes. The initial size of each $\ell_i \in L$ is expressed by its width $w_i > 0$ and height $h_i > 0$. When the *scale factor* is σ , the size of ℓ_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 (Fig. 2 (a)). The anchor

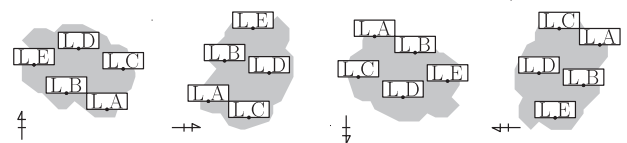


Fig. 1 Example of the label size maximization problem for rotating maps.

A preliminary version was presented at the 25th Canadian Conference on Computational Geometry (CCCG 2013) [24].

¹ Information and System Engineering Course, Graduate School of Science and Engineering, Chuo University, Bunkyo, Tokyo 112-8551, Japan

² Department of Information and System Engineering, Faculty of Science and Engineering, Chuo University, Bunkyo, Tokyo 112-8551, Japan

^{a)} yoyu@imai-lab.ise.chuo-u.ac.jp

^{b)} imai@ise.chuo-u.ac.jp

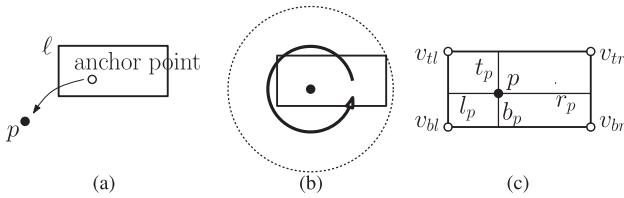


Fig. 2 Definitions.

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

Rectangle shape	MSR	MSBR
Unit squares	$O(n \log n)$	$O(n \log n)$ (Corollary 5)
Squares	(Theorem 4)	$O(n \log n)$ (Theorem 11)
Unit-height rectangles	$O(n \log n)$	$O(n \log n)$ (Corollary 9)
Rectangles	(Theorem 6)	$O(n \log n)$ (1/2-approx., Theorem 13)

point is inside the label ℓ_i or on its boundary. When the label ℓ_i is fixed in place, we say that it is *anchored* at p_i . While M fully rotates from 0 to 2π with the anchor points touching the corresponding points, the labels are placed horizontally, and should not intersect each other. Our problem is to find the *maximum scale factor* σ^* such that the labels do not intersect, and to determine the placing 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 in which 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 for dynamic maps is a natural extension of the label size maximization problem for static maps.

Our results are summarized in **Table 1**. We address several rectangular label shapes (e.g., unit squares and unit-height rectangles). Static label size maximization is NP-hard [9] even if all labels have a unit square shape. However, surprisingly, MSR and MSBR can be solved in polynomial time for several label shapes.

In the following, we treat the clockwise rotation of M as the counterclockwise rotation of the labels around their anchor points (Fig. 2 (b)). Both rotations are equivalent and yield exactly the same results. Gemsa et al. [12] used the same manner.

1.2 Related Work

In map labeling, two models have been considered with respect to the number of label candidates for each point: the *fixed-position model* [9] and the *slider model* [22]. 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- and 4-position models). 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 static label size maximization problems, except for the 1- and 2-position models, are APX-hard, even for unit square labels [9]. Constant-factor approximation algorithms have been obtained for axis-parallel square and rectangular labels [9], [14]. Further, Doddi et al. [7] have examined unit

square labels with different orientations, and Zhu and Qin [26] have considered the case in which all the square labels have the same orientation. Furthermore, static label number maximization problems in several models are known to be NP-hard (e.g., Refs. [9], [22]). Therefore, many approximation algorithms have already been presented (e.g., Refs. [1], [22]).

In dynamic map labeling, Been et al. [2] have provided consistency desiderata, which are that labels should not pop and jump during panning and zooming. Been et al. [3] treated the problems of maximizing sum of active ranges, where the active range of a label ℓ is a contiguous range of the map scales at which ℓ is displayed. Moreover, these problems require that the labels are pairwise disjoint at any scale and satisfy the consistency desiderata. Been et al. have proven that their problems for points in the plane are NP-hard, and presented several exact and approximation algorithms for points in 1D and 2D. Liao et al. [15] have presented several approximation algorithms for several settings of this problem. Gemsa et al. [11] have extended the above problems to the slider model, and have also dealt with selecting the slider positions. Zhang et al. [25] have presented an approximation algorithm for the problem of maximizing the minimum active range. Moreover, Gemsa et al. [12] have considered similar dynamic map labeling for rotating maps. They have also proven that this problem is NP-hard, and presented approximation algorithms. For the same problem, Gemsa et al. [13] compared the approximation algorithm, heuristics, and a deterministic algorithm experimentally. Furthermore, Gemsa et al. [10] extended this problem to the trajectory of a point. Although the above problems have mainly been basically considered for the 1-position model, Buchin and Gerrits [5] showed that dynamic map labeling for 4-position, 2-slider, or 4-slider models is strongly PSPACE-complete.

In the circular labeling problem [19], the corresponding point in the plane is on the boundary of the circular label. However, in MSR and MSBR, during rotating the map, the anchor point is inside the label or on its boundary, and it may not be on the circle obtained by rotating the label.

2. Properties

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

Let ℓ_p be a label anchored at a point p with initial width w_p and initial height h_p . Further, the top-left, top-right, bottom-left, and bottom-right points of ℓ rotated by angle 0 are denoted by v_{tl} , v_{tr} , v_{bl} , and v_{br} , respectively. The segments passing through p are drawn in parallel with the edges of ℓ_p . We assume that p divides the horizontal segment and vertical segment internally with the ratios $l_p : r_p$ (where $r_p = 1 - l_p$) and $t_p : b_p$ (where $b_p = 1 - t_p$), respectively (Fig. 2 (c)). We define each parameter for a point p' in the same way. Thus, $\ell_{p'}$ is the label of p' . $w_{p'}$ and $h_{p'}$ are the initial width and initial height of $\ell_{p'}$, respectively. Moreover, the parameters of $\ell_{p'}$ are defined as v'_{tl} , v'_{tr} , v'_{bl} , v'_{br} , $l_{p'}$, $r_{p'}$, $t_{p'}$ and $b_{p'}$, respectively. Finally, let $d_{pp'}$ be the distance between p and p' , and $\sigma_{pp'}$ be the maximum scale factor for p and p' .

Lemma 1. Let ℓ_p and $\ell_{p'}$ be two labels, which are anchored at

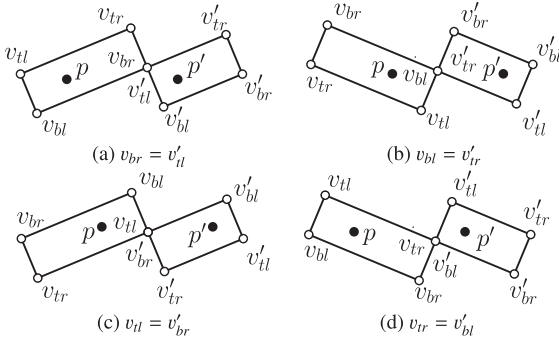


Fig. 3 Four possible cases in which the corner points of the two labels ℓ_p and $\ell_{p'}$ intersect.

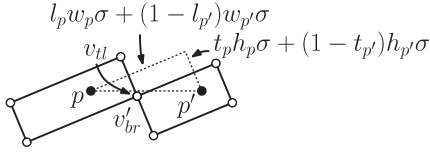


Fig. 4 The case in which $v_{tl} = v'_{br}$.

points p and p' , respectively. ℓ_p and $\ell_{p'}$ can be placed with the maximum scale factor $\sigma_{pp'}$ if and only if the anchor points of ℓ_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 σ be the scale factor for p and p' . Note that ℓ_p and $\ell_{p'}$ touch at their corner points for some rotation θ . Otherwise, if ℓ_p and $\ell_{p'}$ touch on their boundary segments, they overlap by slight rotation. Moreover, ℓ_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 $v_{tr} = v'_{bl}$ (**Fig. 3**).

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

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

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

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

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

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

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

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

First, we focus on the inequalities (1) and (2). As the denominators of the right-hand sides of inequalities (1) and (2) decrease, the maximum possible σ increases. $(t_p h_p + (1 - t_{p'}) h_{p'})^2$ appears on the right-hand sides of both inequalities (1) and (2). The smaller $(l_p w_p + (1 - l_{p'}) w_{p'})^2$ is, the greater $((1 - l_p) w_p + l_{p'} w_{p'})^2$ is. Therefore, if $(l_p w_p + (1 - l_{p'}) w_{p'})^2 = ((1 - l_p) w_p + l_{p'} w_{p'})^2$, σ is maximized among values satisfying inequalities (1) and (2). This condition is equivalent to the equation $(1 - 2l_p)w_p = (1 - 2l_{p'})w_{p'}$. We can obtain the same result for the inequalities (3) and (4).

Similarly, we focus on the inequalities (1) and (4). If $(1 -$

$2t_p)h_p = (1 - 2t_{p'})h_{p'}$, σ is maximized among values satisfying inequalities (1) and (4). The above two equations are satisfied simultaneously. This argument is also satisfied for inequalities (2) and (3). Therefore, if $(1 - 2l_p)w_p = (1 - 2l_{p'})w_{p'}$ and $(1 - 2t_p)h_p = (1 - 2t_{p'})h_{p'}$, σ is the maximum scale factor $\sigma_{pp'}$.

The converse is also true. \square

From Lemma 1, we can obtain the following lemma.

Lemma 2. For given points p and p' with labels ℓ_p and $\ell_{p'}$, respectively, if the anchor point of each label is the intersection of two diagonals of the label, ℓ_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 in which 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 presented in Lemma 1 are satisfied and, hence, the scale factor $\sigma_{pp'}$ is maximized. \square

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

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

Proof. The naive algorithm of MSR gives the maximum scale factor σ^* for the unit-height rectangular labels. We consider the points obtained by translating the anchor points placed at the center of rectangles in MSR to either one of the top and bottom (for unit-width rectangular labels, left and right) boundaries. Those points satisfy inequalities (1)–(4) in Lemma 1. Therefore, those points are the anchor points in MSBR and σ^* is also the maximum scale factor in MSBR. \square

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

3. MSR with Square Labels

When all the labels are squares, the problem has a strong connection to the *weighted closest pair problem* [8]: 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 σ is a scale factor. The goal is to find the maximum scale factor σ^* such that the disks are pairwise disjoint. **Theorem 4.** MSR with square labels can be solved in $O(n \log n)$ time and $O(n)$ space.

Proof. Let p and p' be two points having square labels, and lying 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' so that 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 p and p' . Then, the maximum scale factor $\sigma_{pp'}$ is obtained by maximizing the disk sizes such that they are pairwise disjoint.

Therefore, MSR with square labels is regarded 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 [8] has presented 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 with unit square labels can be solved in $O(n \log n)$ time and $O(n)$ space.*

4. MSR with Rectangular Labels

The algorithm of square labels does not work directly for the rectangular labels 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 [8] used in the weighted closest pair problem can be modified for MSR and MSBR with arbitrary and unit-height rectangular labels, respectively. First, our modified algorithm overestimates the maximum scale factor, and then fixes the maximum value using the intersection graph of the disks drawn by full rotation of the labels. In the algorithm, we use the *Delaunay triangulation* [6], [17] of P , $DT(P)$, which is a triangulation with the *empty circle property*: For any triangle T in $DT(P)$, the circumcircle of T contains no points of P in its interior. We call a triangle of $DT(P)$ a *Delaunay triangle*. When points p and q are vertices of a Delaunay triangle in $DT(P)$, q is called a *neighbor* of p .

Our algorithm can be described as Algorithm 1.

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 $\frac{\sigma_{\text{pre}}}{2} \sqrt{w_p^2 + h_p^2}$. **Theorem 6.** *MSR can be solved in $O(n \log n)$ time and $O(n)$ space.*

In order to prove Theorem 6, we present first 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 $R_p = \frac{\sigma_{\text{pre}}}{2} \sqrt{w_p^2 + h_p^2}$. From the definition of σ_{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 $DT(P)$. Therefore, the radius of D_p is less than $|\overline{pq}|$. Hence, D_p cannot contain points that are not 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 for which the 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 in which 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 $q \neq p, p'$, 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 that the x -coordinate of p' is greater than that of p (Fig. 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 manner. Let s be the intersection of the boundary of D_p and $\overline{pp'}$. In the following, we assume $0 = x_p \leq x_q \leq x_s$ and $0 = y_p = y_{p'} \leq y_q$. q is in the shaded area in Fig. 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 $x_s < x_q \leq x_{p'}$, these cases can be proven in the same manner.

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

Case 1: q is a neighbor of p in $DT(P)$.

We denote $\frac{\sigma_{\text{pre}}}{2} \sqrt{(w_p + w_q)^2 + (h_p + h_q)^2}$ by \widetilde{R}_{pq} . In this case, by the definition of σ_{pre} , we have $|\overline{pq}| \geq \widetilde{R}_{pq}$. 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 Fig. 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 in which $x_q = x_s$. Because $|\overline{pq}| \geq \widetilde{R}_{pq}$ and $|\overline{ps}| = R_p = \frac{\sigma_{\text{pre}}}{2} \sqrt{w_p^2 + h_p^2}$, we have that

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

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

Case 2: q is not a neighbor of p in $DT(P)$.

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 manner. If p' is a neighbor of p in $DT(P)$, the circumcircle of a Delaunay

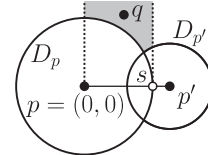


Fig. 5 Assumption of Lemma 8.

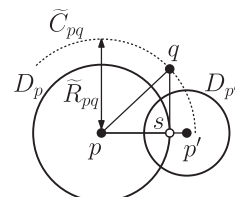


Fig. 6 Case 1 of Lemma 8.

Algorithm 1 Algorithm for MSR.

- 1: Compute $DT(P)$ for P .
- 2: For each point p , calculate the maximum scale factor σ_p with all the neighbors in $DT(P)$. Take the minimum scale factor $\sigma_{\text{pre}} = \min_{p \in P} \sigma_p$ of all the scale factors.
- 3: For each point $p \in P$, draw a closed disk with center p and radius $\frac{\sigma_{\text{pre}}}{2} \sqrt{w_p^2 + h_p^2}$. Enumerate all intersections of disks using the standard intersection detection 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 σ^* .

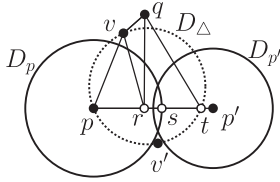


Fig. 7 Case 2-1 of Lemma 8.

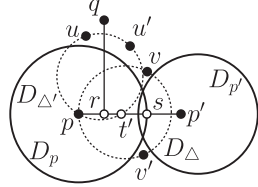


Fig. 8 Case 2-2a of Lemma 8.

triangle that has the edge $\overline{pp'}$ completely contains $\overline{pp'} \cap D_p$. If p' is not a neighbor of p in $DT(P)$, there is a Delaunay triangle $\Delta pvv'$ that has an edge that intersects the inside of $\overline{pp'}$. Since $v, v' \notin D_p$ by Lemma 7, the circumcircle of $\Delta pvv'$ completely contains $\overline{pp'} \cap D_p$. Therefore, we consider such a Delaunay triangle $\Delta pvv'$. In this case, v' might be p' . Without loss of generality, we assume that $y_v > 0$ and $y_{v'} \leq 0$. Further, let t be the intersection point of the circumcircle of $\Delta pvv'$ and $\overline{pp'}$. Since the circumcircle contains $\overline{pp'} \cap D_p$ regardless of whether p' is a neighbor of p in $DT(P)$, we have $x_q \leq x_s < x_t$. In the case $x_v = x_q$, since $y_v < y_q$ and by Lemma 7, D_q cannot intersect $\overline{pp'}$. Therefore, we consider two cases: $x_v < x_q \leq x_s$ and $x_q < x_v < x_s$. In the following, we denote the closed disk whose boundary is the circumcircle of $\Delta pvv'$ by D_Δ .

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

Let r be the intersection of $\overline{pp'}$ and a perpendicular from q to $\overline{pp'}$ (Fig. 7). Since D_Δ completely contains $\overline{pp'} \cap D_p$, $x_r \leq x_s < x_t$. Because $\angle prq = \pi/2$ and $x_t - x_s > 0$, $\angle ptq < \pi/2$. Moreover, since $q \notin D_\Delta$ by the empty circle property, $\angle pvq + \angle ptq \geq \pi$. Therefore, we have $\angle pvq \geq \pi - \angle ptq > \pi/2$. Because $r \in D_p$ and $v \notin D_p$, $|\overline{pr}| < |\overline{pv}|$ and $\angle pvr \leq \angle prv$. 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 $\Delta puu'$ such that $x_u \leq x_q \leq x_{u'}$. Since q is not a neighbor of p in $DT(P)$, there is a Delaunay triangle $\Delta pzz'$ 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 $\Delta pzz'$ to $\Delta pvv'$, there is a Delaunay triangle $\Delta puu'$ that satisfies the condition. In the following, we denote the closed disk whose boundary is the circumcircle of $\Delta puu'$ by $D_{\Delta'}$, and the boundary of $D_{\Delta'}$ by $C_{\Delta'}$. We consider two cases $r \in D_{\Delta'} \setminus C_{\Delta'}$ and $r \notin D_{\Delta'} \setminus C_{\Delta'}$.

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

Let t' be the intersection of $C_{\Delta'}$ and the inside of $\overline{pp'}$ (Fig. 8). Since $r \in D_{\Delta'} \setminus C_{\Delta'}$, we have $x_r < x_{t'}$. Therefore, we can use the same proof as for Case 2-1 by replacing u and t' with v and t , respectively. It is shown that D_q cannot intersect $\overline{pp'}$.

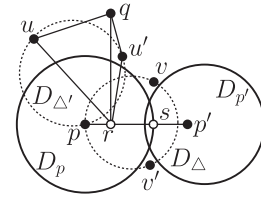


Fig. 9 Case 2-2b of Lemma 8.

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

In this case, we consider the quadrilateral $qu'u'r$ (Fig. 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 Δqur , we have $\angle qur < \pi/2$. Since q and r are outside $D_{\Delta'}$, 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 manner. Therefore, D_q cannot intersect $\overline{pp'}$. \square

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

Next, we show the complexity. Step 1 can be computed in $O(n \log n)$ time and $O(n)$ space [6], [17]. Step 2 calculates the maximum scale factor between neighbors in $DT(P)$. 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. \square

Corollary 9. MSBR with unit-height (or unit-width) rectangular labels can be solved 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 bounded by $\Omega(n \log n)$ [17], which may also apply to our problems.

5. MSBR with Square Labels

In this section, we describe the $O(n \log n)$ -time algorithm for MSBR with arbitrary square labels (MSBRwS). First, we show locations of the anchor points when the scale factor is maximized for this problem.

Lemma 10. For given points p and p' with respective labels ℓ_p and $\ell_{p'}$, if the anchor point of each label is at the center of the left side of the label, ℓ_p and $\ell_{p'}$ can be placed using the maximum scale factor $\sigma_{pp'}$ for MSBRwS.

Proof. In this problem, since the width w_p of the label ℓ_p is identical to the height h_p , we denote the height as w_p . Similarly, the height of the label $\ell_{p'}$ is denoted as $w_{p'}$. Without loss of generality, we assume that p and p' lie on a horizontal line, that the

x -coordinate of p' is greater than that of p , and that w_p is less than or equal to $w_{p'}$.

In MSBR, the anchor points must lie on the boundaries of their labels. Therefore, at least one of the conditions $l_p = 0$, $l_p = 1$, $t_p = 0$, and $t_p = 1$ is satisfied. Moreover, for square labels, the four cases are equivalent, because each case can be transformed to the other cases by rotating the map. Therefore, we consider $l_p = 0$ only. Similarly, for the label $\ell_{p'}$, at least one of the conditions $l_{p'} = 0$, $l_{p'} = 1$, $t_{p'} = 0$, and $t_{p'} = 1$ is satisfied. Therefore, we consider the following four possible cases.

Case 1: $l_p = 0$ and $l_{p'} = 0$.

In this case, inequalities (1)–(4) can be expressed as follows:

$$\sigma \leq d_{pp'} / \sqrt{w_{p'}^2 + (t_p w_p + (1-t_{p'})w_{p'})^2}, \quad (5)$$

$$\sigma \leq d_{pp'} / \sqrt{w_p^2 + (t_p w_p + (1-t_{p'})w_{p'})^2}, \quad (6)$$

$$\sigma \leq d_{pp'} / \sqrt{w_p^2 + ((1-t_p)w_p + t_{p'}w_{p'})^2}, \quad (7)$$

$$\sigma \leq d_{pp'} / \sqrt{w_{p'}^2 + ((1-t_p)w_p + t_{p'}w_{p'})^2}. \quad (8)$$

Since $w_p < w_{p'}$, we consider inequalities (5) and (8) only. If $(1-2t_p)w_p = (1-2t_{p'})w_{p'}$, σ is maximized among the values that satisfy inequalities (5)–(8) in the same manner as for Lemma 1. Therefore, in this case, the maximum scale factor is

$$d_{pp'} / \sqrt{w_{p'}^2 + (\frac{1}{2}(w_p + w_{p'}))^2}.$$

Case 2: $l_p = 0$ and $l_{p'} = 1$.

Inequalities (1)–(4) are expressed as follows:

$$\sigma \leq d_{pp'} / \sqrt{(t_p w_p + (1-t_{p'})w_{p'})^2},$$

$$\sigma \leq d_{pp'} / \sqrt{(w_p + w_{p'})^2 + (t_p w_p + (1-t_{p'})w_{p'})^2},$$

$$\sigma \leq d_{pp'} / \sqrt{(w_p + w_{p'})^2 + ((1-t_p)w_p + t_{p'}w_{p'})^2},$$

$$\sigma \leq d_{pp'} / \sqrt{((1-t_p)w_p + t_{p'}w_{p'})^2}.$$

In the same manner as for Case 1, σ is maximized when $(1-2t_p)w_p = (1-2t_{p'})w_{p'}$. Furthermore, the maximum scale factor is

$$d_{pp'} / \sqrt{(w_p + w_{p'})^2 + (\frac{1}{2}(w_p + w_{p'}))^2}.$$

Case 3: $l_p = 0$ and $t_{p'} = 0$.

Inequalities (1)–(4) are expressed as follows:

$$\sigma \leq d_{pp'} / \sqrt{((1-l_{p'})w_{p'})^2 + (t_p w_p + w_{p'})^2}, \quad (9)$$

$$\sigma \leq d_{pp'} / \sqrt{(w_p + l_{p'}w_{p'})^2 + (t_p w_p + w_{p'})^2}, \quad (10)$$

$$\sigma \leq d_{pp'} / \sqrt{(w_p + l_{p'}w_{p'})^2 + ((1-t_p)w_p)^2}, \quad (11)$$

$$\sigma \leq d_{pp'} / \sqrt{((1-l_{p'})w_{p'})^2 + ((1-t_p)w_p)^2}. \quad (12)$$

First, we focus on inequalities (9) and (10). If $((1-l_{p'})w_{p'})^2 = (w_p + l_{p'}w_{p'})^2$, σ is maximized among values that satisfy inequalities (9) and (10). This condition is equivalent to the equation $l_{p'} = (1-w_p/w_{p'})/2$. Similarly, for the case in which inequalities (9) and (12) are satisfied, if $t_p = (1-w_{p'}/w_p)/2$, σ is maximized among values that satisfy inequalities (9) and (12). Since $w_p \leq w_{p'}$ from the hypothesis, we consider the cases $w_p = w_{p'}$ and $w_p < w_{p'}$ separately. If $w_p = w_{p'}$, $t_p = 0$. Otherwise, t_p becomes negative, which is impossible. Therefore, even for

Algorithm 2 Algorithm for MSBRWS.

- 1: Compute $DT(P)$ for P .
 - 2: For each point p , calculate the maximum scale factor σ_p with all the neighbors in $DT(P)$. Take the minimum scale factor $\sigma_{pre} = \min_{p \in P} \sigma_p$ of all the scale factors.
 - 3: For each point $p \in P$, draw a closed disk with center p and radius $\frac{\sqrt{5}}{2}w_p\sigma_{pre}$. Enumerate all intersections of disks using the standard intersection detection 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 σ^* .
-

this case, σ is maximized if $t_p = 0$. Since the above two conditions are satisfied simultaneously, the maximum scale factor is

$$d_{pp'} / \sqrt{(\frac{1}{2}(w_p + w_{p'}))^2 + w_{p'}^2}.$$

Case 4: $l_p = 0$ and $t_{p'} = 1$.

In the same manner as that used for case 3, if $l_{p'} = (1-w_p/w_{p'})/2$ and $t_p = 1$, σ is maximized and the value is

$$d_{pp'} / \sqrt{(\frac{1}{2}(w_p + w_{p'}))^2 + w_{p'}^2}.$$

From above, the maximum scale factors of Cases 1, 3, and 4 are identical, and are greater than that of Case 2. Therefore, in Cases 1, 3, and 4, there are anchor points that maximize the scale factor among all possible cases. Furthermore, in Case 1, the equation of the anchor points, that maximizes the scale factor contains $t_p = t_{p'} = 1/2$. This completes the proof. \square

From Lemma 10, the maximum scale factor of MSBRWS for two points p and p' is $d_{pp'} / \sqrt{w_{p'}^2 + (\frac{1}{2}(w_p + w_{p'}))^2}$, and we can solve MSBRWS for more than two points using the naïve $\Theta(n^2)$ -time algorithm, as described in Section 2. In the following, we present the $O(n \log n)$ -time algorithm.

The $O(n \log n)$ -time algorithm can be described as shown in Algorithm 2, and is mostly identical to Algorithm 1. The difference is that the maximum scale factor is calculated for two points in Steps 2 and 4 using the anchor points obtained in Lemma 10, and the radius of each disk in Step 3, where the radii are also obtained from Lemma 10.

Theorem 11. *Algorithm 2 can solve MSBRWS in $O(n \log n)$ time and $O(n)$ space.*

Proof. Essentially, the proof of this theorem is identical to that for Theorem 6. In Algorithm 2, however, Lemma 8 does not hold. Therefore, the following shows that the number of intersecting pairs obtained at Step 3 of Algorithm 2 is $O(n)$.

In order to demonstrate this point, we use the *sphere-of-influence graph (SIG)* [21] of P . The vertex set of SIG is P . The edge set consists of edges connecting two vertices whose nearest-neighbor circles intersect. The nearest-neighbor circle of point p is a circle centered at p , and the radius is the distance from p to its nearest-neighbor. Soss [18] has proven that the number of edges in SIG is at most $15n$.

As in Lemma 8, we consider the straight-line graph constructed by connecting two points, if the disks intersect at Step 3 of Algorithm 2. From Lemma 7, each disk is smaller than the nearest-neighbor circle. Therefore, the number of edges in the graph is also at most $15n$, and this proof is completed. \square

The above proof can be also applied to Algorithm 1. However, the proof in Theorem 6 indicates that MSR and MSBR with

unit-height rectangular labels can be solved faster than MSBRwS actually. Furthermore, as an alternative approach to solve MSBR, we can consider the following algorithm: First, construct SIG from P . Then, for each edge of SIG, calculate the maximum scale factor for its end points and choose the minimum among them. The algorithm to construct SIG, which is not described precisely [20], may be similar with Algorithm 2. Therefore, we think our algorithm can solve MSBRwS faster than the alternative algorithm actually because the number of edges in the intersection graph constructed in the proof of Theorem 11 is actually less than the number of edges in SIG.

6. MSBR with Rectangular Labels

In this section, we describe the $1/2$ -approximation algorithm for MSBR with arbitrary rectangular labels. First, we show the locations of anchor points using the algorithm, and then we describe the algorithm itself, which is also mostly identical to Algorithm 1.

Lemma 12. *For all points, the anchor point of each label is placed at the center of the left side of the label. Then, the maximum scale factor for this placement is greater than half the maximum scale factor for MSBR.*

Proof. First, we consider MSBR for two points p and p' . Without loss of generality, we assume $w_p \leq w_{p'}$. If the anchor point for each label is at the center of its left side, inequalities (1)–(4) are expressed as follows:

$$\begin{aligned}\sigma &\leq d_{pp'} / \sqrt{w_{p'}^2 + \left(\frac{1}{2}(h_p + h_{p'})\right)^2}, \\ \sigma &\leq d_{pp'} / \sqrt{w_p^2 + \left(\frac{1}{2}(h_p + h_{p'})\right)^2}.\end{aligned}$$

Since $w_p \leq w_{p'}$, $d_{pp'} / \sqrt{w_{p'}^2 + (\frac{1}{2}(h_p + h_{p'}))^2}$ is the maximum scale factor for two points. For more than two points, therefore, the maximum scale factor is $\min \left\{ d_{pp'} / \sqrt{w_{p'}^2 + (\frac{1}{2}(h_p + h_{p'}))^2} \mid p, p' \in P \right\}$, if the anchor point of each label is at the center of its left side. We denote this value by σ_{LC} .

Furthermore, from Section 2, the maximum scale factor σ_{MSR} for MSR with arbitrary rectangular labels is

$$\min \left\{ 2d_{pp'} / \sqrt{(w_p + w_{p'})^2 + (h_p + h_{p'})^2} \mid p, p' \in P \right\}.$$

Since the maximum scale factor σ_{MSBR} is less than or equal to σ_{MSR} ,

$$\begin{aligned}\sigma_{LC} &= \min \left\{ d_{pp'} / \sqrt{w_{p'}^2 + \left(\frac{1}{2}(h_p + h_{p'})\right)^2} \mid p, p' \in P \right\} \\ &> \min \left\{ d_{pp'} / \sqrt{(w_p + w_{p'})^2 + (h_p + h_{p'})^2} \mid p, p' \in P \right\} \\ &= \frac{1}{2}\sigma_{MSR} \geq \frac{1}{2}\sigma_{MSBR}.\end{aligned}$$

Therefore, this proof is completed. \square

By using the above lemma and modifying Algorithm 1 slightly, we can obtain a $1/2$ -approximation algorithm for MSBR as described in Algorithm 3. As for Algorithm 2, the difference compared to Algorithm 1 is the use of anchor points in Steps 2 and 4,

Algorithm 3 $1/2$ -approximation algorithm for MSBR.

- 1: Compute $DT(P)$ for P .
 - 2: For each point p , calculate the maximum scale factor σ_p with all the neighbors in $DT(P)$. Take the minimum scale factor $\sigma_{pre} = \min_{p \in P} \sigma_p$ of all the scale factors.
 - 3: For each point $p \in P$, draw a closed disk with center p and radius $\sigma_{pre} \sqrt{w_p^2 + (\frac{1}{2}h_p)^2}$. Enumerate all intersections of disks using the standard intersection detection 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 σ^* .
-

and the radius of each disk in Step 3.

From Lemma 12 and the arguments in Theorem 11, the following theorem can be obtained.

Theorem 13. *Algorithm 3 is a $1/2$ -approximation, $O(n \log n)$ -time, and $O(n)$ -space algorithm for MSBR.*

7. 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 can be solved in polynomial time for rotating maps, and we presented efficient algorithms for finding the maximum scale factor.

The most interesting open question is whether MSBR with arbitrary rectangular labels can be solved in polynomial time. Another further research is label size maximization problems for other dynamic maps (e.g., zooming map, the trajectory of a point).

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] Agarwal, P.K., van Kreveld, M.J. and Suri, S.: Label placement by maximum independent set in rectangles, *Comput. Geom.*, Vol.11, No.3-4, pp.209–218 (1998).
- [2] Been, K., Daiches, E. and Yap, C.-K.: Dynamic Map Labeling, *IEEE Trans. Vis. Comput. Graph.*, Vol.12, No.5, pp.773–780 (2006).
- [3] Been, K., Nöllenburg, M., Poon, S.-H. and Wolff, A.: Optimizing active ranges for consistent dynamic map labeling, *Comput. Geom.*, Vol.43, No.3, pp.312–328 (2010).
- [4] Bentley, J.L. and Ottmann, T.: Algorithms for Reporting and Counting Geometric Intersections, *IEEE Trans. Computers*, Vol.28, No.9, pp.643–647 (1979).
- [5] Buchin, K. and Gerrits, D.H.P.: Dynamic Point Labeling is Strongly PSPACE-Complete, *Int. J. Comput. Geometry Appl.*, Vol.24, No.4, pp.373–395 (2014).
- [6] de Berg, M., Cheong, O., van Kreveld, M. and Overmars, M.: *Computational Geometry: Algorithms and Applications*, 3rd edition, Springer-Verlag (2008).
- [7] Doddi, S., Marathe, M.V., Mirzaian, A., Moret, B.M.E. and Zhu, B.: Map Labeling and Its Generalizations, *Proc. 8th Annual ACM-SIAM Symposium on Discrete Algorithms*, Saks, M.E. (Ed.), pp.148–157, ACM/SIAM (1997).
- [8] Formann, M.: Weighted Closest Pairs, *STACS 93, Proc. 10th Annual Symposium on Theoretical Aspects of Computer Science*, Enjalbert, P., Finkel, A. and Wagner, K.W. (Eds.), Lecture Notes in Computer Science, Vol.665, pp.270–281, Springer (1993).
- [9] Formann, M. and Wagner, F.: A Packing Problem with Applications to Lettering of Maps, *Proc. 7th Annual Symposium on Computational Geometry*, Drysdale, R.L.S. (Ed.), pp.281–288, ACM (1991).
- [10] Gamsa, A., Niedermann, B. and Nöllenburg, M.: Trajectory-Based Dynamic Map Labeling, *Algorithms and Computation - Proc. 24th International Symposium, ISAAC 2013*, Cai, L., Cheng, S. and Lam, T.W. (Eds.), Lecture Notes in Computer Science, Vol.8283, pp.413–423, Springer (2013).
- [11] Gamsa, A., Nöllenburg, M. and Rutter, I.: Sliding labels for dynamic

- point labeling, *Proc. 23rd Annual Canadian Conference on Computational Geometry* (2011).
- [12] Gamsa, A., Nöllenburg, M. and Rutter, I.: Consistent Labeling of Rotating Maps, *JoCG*, Vol.7, No.1, pp.308–331 (2016).
- [13] Gamsa, A., Nöllenburg, M. and Rutter, I.: Evaluation of Labeling Strategies for Rotating Maps, *ACM Journal of Experimental Algorithms*, Vol.21, No.1, pp.1.4:1–1.4:21 (2016).
- [14] Jung, J.-W. and Chwa, K.-Y.: Labeling points with given rectangles, *Inf. Process. Lett.*, Vol.89, No.3, pp.115–121 (2004).
- [15] Liao, C., Liang, C. and Poon, S.-H.: Approximation algorithms on consistent dynamic map labeling, *Theor. Comput. Sci.*, Vol.640, pp.84–93 (2016).
- [16] Poon, S.-H. and Shin, C.-S.: Adaptive Zooming in Point Set Labeling, *15th International Symposium on Fundamentals of Computation Theory (FCT 2005)*, pp.222–233 (2005).
- [17] Preparata, F.P. and Shamos, M.I.: *Computational Geometry - An Introduction*, Springer (1985).
- [18] Soss, M.A.: On the size of the Euclidean sphere of influence graph, *Proc. 11th Canadian Conference on Computational Geometry* (1999).
- [19] Strijk, T. and Wolff, A.: Labeling Points with Circles, *Int. J. Comput. Geometry Appl.*, Vol.11, No.2, pp.181–195 (2001).
- [20] Toussaint, G.T.: A Graph-Theoretical Primal Sketch, *Computational Morphology, A Computational Geometric Approach to the Analysis of Form*, Toussaint, G.T. (Ed.), pp.222–260 (1988).
- [21] Toussaint, G.T.: The Sphere of Influence Graph: Theory and Applications, *International Journal of Information Technology & Computer Science*, Vol.14, No.2, pp.37–42 (2014).
- [22] van Kreveld, M.J., Strijk, T. and Wolff, A.: Point labeling with sliding labels, *Comput. Geom.*, Vol.13, No.1, pp.21–47 (1999).
- [23] Wolff, A. and Strijk, T.: The Map-Labeling Bibliography (2009), available from (<http://i11www.iti.uni-karlsruhe.de/map-labeling/bibliography>).
- [24] Yokosuka, Y. and Imai, K.: Polynomial Time Algorithms for Label Size Maximization on Rotating Maps, *Proc. 25th Canadian Conference on Computational Geometry, CCCG 2013* (2013).
- [25] Zhang, X., Poon, S.-H., Li, M. and Lee, C.S.V.: On Maxmin Active Range Problem for Weighted Consistent Dynamic Map Labeling, *The 7th International Conference on Advanced Geographic Information Systems, Applications, and Services*, pp.32–37 (2015).
- [26] Zhu, B. and Qin, Z.: New Approximation Algorithms for Map Labeling with Sliding Labels, *J. Comb. Optim.*, Vol.6, No.1, pp.99–110 (2002).



Yusuke Yokosuka received his B.E. and M.E. degrees in Engineering from Chuo University in 2004 and 2006, respectively. Since 2006, he has been with Mitsubishi Electric Corporation. He is also a Ph.D. candidate at Chuo University from 2012. His research interests include algorithm theory, computational geometry, and their

applications.



Keiko Imai received her B.S., M.S., and Dr.S. degrees in Mathematics from Tsuda College in 1980, 1982 and 1991, respectively. Since 1992 she has been with Chuo University, where she is currently as a full professor in the Department of Information and System Engineering. Her research interests include algorithm theory,

computational geometry, discrete mathematics, and their applications. She is a member of IPSJ, JSIAM, IEICE, OR Soc. Japan, Math. Soc. Japan, GIS Association in Japan, and ACM. She received the title of the Fellow of IEICE in 2015.