Covering a Point Set by Two Disjoint Rectangles*

Sang-Sub Kim† Sang Won Bae‡ Hee-Kap Ahn§

Abstract

Given a set S of n points in the plane, the disjoint two-rectangle covering problem is to find a pair of disjoint rectangles such that their union contains S and the area of the larger rectangle is minimized. In this paper we consider two variants of this optimization problem: (1) the rectangles are free to rotate but must remain parallel to each other, and (2) one rectangle is axis-parallel but the other rectangle is allowed to be in arbitrary orientation. For both of the problems, we present $O(n^2 \log n)$-time algorithms using $O(n)$ space.

1 Introduction

For a set S of n points in the plane, the *disjoint two-rectangle covering problem* is to find a pair of disjoint rectangles with arbitrary orientations such that the union of the rectangles contains all the points in S and the area of the larger rectangle is minimized. This is a fundamental optimization problem that deals with covering a point set S in the plane by two geometric objects of the same type. The surveys by Agarwal and Sharir [1] and by Segal [13] provide comprehensive reviews on a list of such problems.

More specifically, the disjoint two-rectangle covering is a generalization of the axis-parallel two-rectangle covering problem in which the two rectangles are restricted to be axis-parallel. Bespamyatnikh and Segal [3] studied the restricted version of the problem and presented a simple $O(n \log n)$ time algorithm that finds the optimal axis-parallel covering. They also extended the result into higher dimensions and presented an $O(n \log n + n^{d-1})$ time algorithm for the problem in d-dimensional space.

For arbitrary orientations, Jaromczyk and Kowaluk [7] gave an $O(n^2)$ time algorithm for the two-square covering problem with the restriction that the two squares are congruent and parallel to each other. Later, Katz, Kedem and Segal [8] considered the discrete rectilinear two-center problem: find two squares covering the point set S such that their centers are constrained to be at points in S and the area of the larger square is minimized. They presented algorithms for three variants of this problem: when two squares are axis-parallel, an $O(n \log^2 n)$-time/$O(n)$-space algorithm; when two squares are parallel to each other but not necessarily axis-parallel, an $O(n^2 \log^4 n)$-time/$O(n^2)$-space algorithm; when both can rotate independently, an $O(n^3 \log^2 n)$-time/$O(n^2)$-space algorithm. Recently Saha and Das considered the two-rectangle covering problem with restriction that the two rectangles are parallel to each other, and presented an algorithm that finds an optimal two-rectangle covering in time $O(n^3)$ using $O(n^2)$ space [11].

*This work was supported by the Korea Research Foundation Grant funded by the Korean Government (MOEHRD, Basic Research Promotion Fund) (KRF-2007-331-D00372) and by the Brain Korea 21 Project.

†Graduate School for Information Technology, POSTECH, South Korea. helmet1981@postech.ac.kr

‡Division of Computer Science, KAIST, South Korea. swbae@tclab.kaist.ac.kr

§Department of Computer Science and Engineering, POSTECH, South Korea. heekap@postech.ac.kr

In this paper, we present an $O(n^2 \log n)$ time algorithm that finds an optimal disjoint parallel two-rectangle covering of S in arbitrary orientation, which improves the result of Saha and Das [11]. We also consider a variant of the problem in which one rectangle is axis-parallel and the other is allowed to have an arbitrary orientation while they remain disjoint. We present an $O(n^2 \log n)$ time algorithm for this variant. Moreover, both of our algorithms presented in this paper use only linear space. The general approach to most of optimal covering problems is first to solve the corresponding decision problem, then to apply an optimization scheme, such as the sorted matrices technique [5], the expander-based technique [9], or parametric search [10]. In contrast, our algorithms are rather intuitive: based on a few geometric observations and analysis of the area functions of rectangles, they capture combinatorial changes in the configuration carefully and maintain the optimal two-rectangle covering during rotation.

We also study another variant of the problem in which both rectangles are allowed to have arbitrary orientations independently while they remain disjoint. However, it seems that the same approach does not apply to this generalized case; this is mainly because the area functions are too complicated to analyze. We discuss about this issue at the end of the paper.

2 Covering Points by Two Disjoint Parallel Rectangles

Throughout this paper, a (directed) line ℓ or a rectangle is called *θ-oriented* if it is parallel (or directed) to an orientation θ. We denote by $B_\theta(P)$ the θ-oriented bounding box of a point set P.

2.1 Characterization

Let S be a set of n points in the plane. We assume that no three points lie on a line. For a fixed orientation θ, let ℓ be a θ-oriented directed line which partitions S into two subsets L_ℓ and R_ℓ, where L_ℓ contains the points in S lying in the left side of ℓ and R_ℓ contains the points in S lying in the right side of ℓ; there can be at most two points lying on ℓ and each of them can belong to either L_ℓ or R_ℓ.

Then the optimal disjoint two-rectangle covering problem with restriction to θ is to find a θ-oriented line ℓ such that $\max\{|B_\theta(L_\ell)|, |B_\theta(R_\ell)|\}$ is minimized, where $|\cdot|$ returns the area of a given rectangle. We call such a line ℓ an *optimal partitioning line* of S in θ. We denote by $f(\theta)$ the optimal objective value for fixed orientation θ, that is, $f(\theta) = \min_\ell \max\{|B_\theta(L_\ell)|, |B_\theta(R_\ell)|\}$ for all θ-oriented directed lines ℓ.

The exact value of $f(\theta)$ can be computed in $O(n \log n)$ time [3]; once the points in S are sorted in direction $\theta + \pi/2$, we can find an optimal partitioning line in linear time by using a plane sweep algorithm over S in the direction.

Obviously, there can be infinitely many optimal partitioning lines. Our algorithm implicitly maintains an optimal partitioning line $\ell(\theta)$ which is uniquely defined for any $\theta \in [0, \pi)$. For the purpose, we consider a bit larger rectangles. For a θ-oriented directed line ℓ, let $B_L(\ell)$ and $B_R(\ell)$ be the minimum θ-oriented rectangles such that both have one side on ℓ while $B_L(\ell)$ covers L_ℓ and $B_R(\ell)$ covers R_ℓ. If we sweep the plane by ℓ in direction $\theta + \pi/2$, it is easy to see that $|B_L(\ell)|$ is monotonically decreasing and $|B_R(\ell)|$ is monotonically increasing. Note that $|B_L(\ell)|$ and $|B_R(\ell)|$ are discontinuous during the plane sweep, but the discontinuity occurs only when ℓ sweeps a point in S. We call ℓ a *bisecting line* in θ if $\max\{|B_L(\ell)|, |B_R(\ell)|\}$ is minimized over all such θ-oriented lines, and denote it by $\ell(\theta)$. For each orientation θ, $\ell(\theta)$ is uniquely determined because of the monotonicity of $|B_L(\ell)|$ and $|B_R(\ell)|$. For simplicity of discussion,

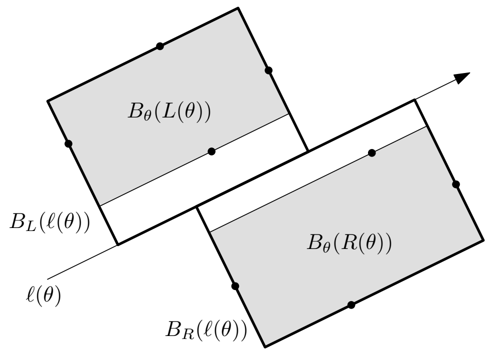

Figure 1: For an orientation θ, the θ-oriented bisecting line $\ell(\theta)$ and its corresponding bounding boxes. Shaded rectangles are the θ-oriented bounding boxes of $L(\theta)$ and of $R(\theta)$, and rectangles with thick sides are $B_L(\ell(\theta))$ and $B_R(\ell(\theta))$.

we let $L(\theta) := L_{\ell(\theta)}$ and $R(\theta) := R_{\ell(\theta)}$. Figure 1 shows the θ-oriented bisecting line $\ell(\theta)$ and the corresponding bounding boxes.

In the following, we show that the bisecting line $\ell(\theta)$ is indeed an optimal partitioning line.

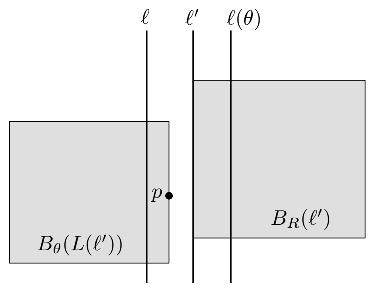

Figure 2: An illustration to the proof of Lemma 1.

Lemma 1 *The θ-oriented bisecting line is an optimal partitioning line in θ.*

Proof. Assume to the contrary that $\ell(\theta)$ is not an optimal partitioning line, and let ℓ be an optimal partitioning line in orientation θ that lies strictly to the left of $\ell(\theta)$. This means that there are some points from R_ℓ lying between them or on the lines, and let p be the point closest to ℓ among those points.

Consider the case that p lies on $\ell(\theta)$. Any θ-oriented line through p is an optimal partitioning line because we can partition S into the same subsets L_ℓ and R_ℓ along the line (Note that a point on a partitioning line can belong either of sides.) Since $\ell(\theta)$ is also a θ-oriented line through p, $\ell(\theta)$ is an optimal partitioning line.

Now consider the case that p lies strictly to the left of $\ell(\theta)$. Then we can always choose a θ-oriented line ℓ' in between p and $\ell(\theta)$ (See Figure 2.) By the definition of the bisecting line, we have $\max\{|B_L(\ell')|, |B_R(\ell')|\} > \max\{|B_L(\ell(\theta))|, |B_R(\ell(\theta))|\}$. Since $|B_L(\ell')| < |B_L(\ell(\theta))|$ by the monotonicity of the area function, $|B_R(\ell')|$ must be larger than $|B_L(\ell')|$.

However, p lies in the left side of ℓ', therefore $|B_\theta(R_\ell)| > |B_R(\ell')| > |B_L(\ell')|$. This implies that $|B_\theta(R_\ell)| > \max\{|B_\theta(L_{\ell'})|, |B_\theta(R_{\ell'})|\}$ and thus we get a contradiction to the optimality of the partitioning line ℓ. $\square$

Consider now that we are allowed to change the orientation θ. Then the optimal disjoint two-rectangle covering problem is to minimize $f(\theta)$ over $\theta \in [0, \pi)$. Before we continue further, we need the following lemma.

Lemma 2 (Saha and Das [11] and Bae et al. [2]) *Let P be a finite set of points in the plane and $(\alpha, \beta) \subset [0, 2\pi)$ be an orientation interval where the sequence of the points touching the sides of $B_\theta(P)$ remains the same for any $\theta \in (\alpha, \beta)$. Then the area of $B_\theta(P)$ can be expressed as a sinusoidal function of θ with angular frequency 2. That is, $|B_\theta(P)|$ is of the form $c_1 \sin(2\theta + c_2) + c_3$, where c_1, c_2, and c_3 are constants depending only on the points on the sides of $B_\theta(P)$.*

On the other hand, we can describe the θ-oriented bounding box $B_\theta(P)$ of a point set P by the sequence of the four touching points, one for each side. Therefore, the optimal two-rectangle covering can be described by a sequence of eight touching points, four from the θ-oriented bounding box of $L(\theta)$ followed by the other four from the other bounding box. We denote by D_θ the sequence of eight touching points at orientation θ and call the points the *determinators* of the two bounding boxes.

During the rotation, we encounter a number of changes in D_θ, which are captured by *events* of following two types:

1. a point in $L(\theta)$ moves into $R(\theta)$ or a point in $R(\theta)$ moves into $L(\theta)$, or

2. a side of $B_\theta(L(\theta))$ or $B_\theta(R(\theta))$ touches two points.

We call an event of the first type a *crossing* event, and an event of the second type a *non-crossing* event. It is not difficult to see why we call such an event either crossing or a non-crossing: conceptually speaking, a point moves from one set to the other set by crossing the bisecting line.

Figure 3 shows how a crossing event occurs during rotation:

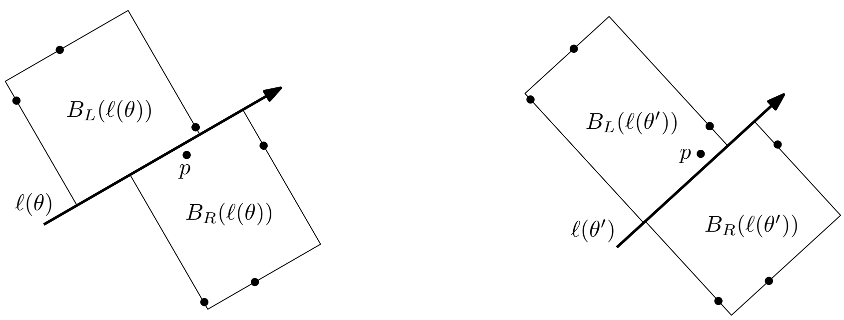

Figure 3: A crossing event occurs during the rotation from θ to $\theta'(> \theta)$: the point p lies in the right side of the bisecting line $\ell(\theta)$ at θ but it lies in the left side of $\ell(\theta')$.

2.2 Algorithm

Our algorithm works by maintaining $L(\theta)$ and $R(\theta)$ as θ increases continuously from 0 to π and minimizing the objective function in orientation intervals where no event occurs.

Algorithm *ParallelTwoRectangleCover(S)*

```
(* computes the optimal parallel two-rectangle covering over orientations in [0, π) *)
θ ← 0 and compute L(0), R(0), and D_0
while θ < π
    do compute the next non-crossing event at θ_n(> θ), assuming no crossing event
       compute the next crossing event at θ_c in [θ, θ_n), if any
       if θ_c is determined, then θ' ← θ_c; otherwise, θ' ← θ_n
       minimize f(ϑ) in the interval [θ, θ')
       θ ← θ'
       update D_θ, L(θ) and R(θ)
return the minimum objective value with its orientation
```

Non-crossing events A non-crossing event corresponds to an event when two points of $L(\theta)$ (or $R(\theta)$) lie on a side of $B_\theta(L(\theta))$ ($B_\theta(R(\theta))$, respectively). Hence, assuming no further crossing event, the next non-crossing event after θ can be computed in constant time once we know the convex hulls of $L(\theta)$ and of $R(\theta)$, and the determinators D_θ as in the well-known rotating caliper technique [14]. For efficient handling of non-crossing events, we make use of the dynamic convex hull structure by Brodal and Jacob [4]. It maintains the convex hull of the points in the plane under insertion and deletion of points in amortized $O(\log n)$ time using $O(n)$ space. So when we update the invariants either by a non-crossing event or by a crossing event, we also update the convex hulls of $L(\theta)$ and $R(\theta)$ in amortized $O(\log n)$ time. Also, we can easily bound the number of non-crossing events during the algorithm: when a non-crossing event occurs at θ, two points in S lie on a line which is either θ-oriented or $(\theta + \pi/2)$-oriented. Since we increase θ from 0 to π, the number of non-crossing events to be handled in the algorithm is $O(n^2)$.

Lemma 3 *The number of non-crossing events is at most $O(n^2)$.*

Minimizing the objective function It can be done in constant time to minimize $f(\vartheta) = \max\{|B_\vartheta(L(\vartheta))|, |B_\vartheta(R(\vartheta))|\}$ in domain $[\theta, \theta')$ where no event occurs due to some nice property of the area functions. The following lemma states such property of the sinusoidal functions of a certain form.

Lemma 4 *A sinusoidal function of the form $c_1 \sin(2\theta + c_2) + c_3$ in the domain $[0, \pi)$ has at most two local minima, where c_1, c_2, and c_3 are constants. For any pair of sinusoidal functions a and b, the equation $a(\vartheta) = b(\vartheta)$ has at most two zeros in the domain unless those functions are the same.*

As observed in Lemma 2, both $|B_\vartheta(L(\vartheta))|$ and $|B_\vartheta(R(\vartheta))|$ are of the form stated in Lemma 4. Thus, $f(\vartheta)$ can be expressed by at most three such pieces of functions, and hence $f(\vartheta)$ can be minimized in constant time in the domain $[\theta, \theta')$.

2.3 Crossing events

What remains is to show how to compute the crossing events, and to bound the number of the events. Before we proceed, we need the following lemma.

Lemma 5 *The bisecting line $\ell(\vartheta)$ moves continuously as ϑ increases from 0 to π.*

Proof. Let $t_L(\vartheta) := |B_L(\ell(\vartheta))|$, $t_R(\vartheta) := |B_R(\ell(\vartheta))|$, and $t(\vartheta) := \max\{t_L(\vartheta), t_R(\vartheta)\}$ be functions of $\vartheta \in [0, 2\pi)$. First, observe that if t is continuous at every $\theta \in (\alpha, \beta)$ for an interval

$(\alpha, \beta) \subset [0, 2\pi)$, then the bisecting line $\ell(\vartheta)$ moves continuously over (α, β). Otherwise, if t is continuous at θ but $\ell(\theta)$ "jumps" at θ, we have two bisecting lines at θ, a contradiction to the uniqueness of the bisecting line.

The main part of the proof shows that the possible number of jumps of $\ell(\vartheta)$ is bounded by a finite number, that is, $\ell(\vartheta)$ can be seen as a piecewise continuous function, which directly implies the lemma since its left and right limits are well defined at everywhere and if they are distinct at some $\theta \in [0, \pi)$, we get two bisecting lines parallel to θ and thus a contradiction to the uniqueness of the bisecting line.

The piecewise continuity of $\ell(\vartheta)$ is shown through the following claim: *if no two points lie on $\ell(\theta)$ for $\theta \in (\alpha, \beta)$, t is continuous in (α, β).* It is easy to see that if $L(\theta)$ and $R(\theta)$ remain the same for every $\theta \in (\alpha, \beta)$, both t_L and t_R are continuous in (α, β). Thus, we are done by checking the continuity of t at θ such that $L(\theta - \varepsilon) \neq L(\theta + \varepsilon)$ for sufficiently small $\varepsilon > 0$ and exactly one point $p \in S$ lies on $\ell(\theta)$. Without loss of generality, we assume that $p \in L(\theta)$ and $p \in R(\theta + \varepsilon)$ so that p crosses over the bisecting line at θ from left to right. Then, we have

$$t_R(\theta) \leq \lim_{\vartheta \to \theta^+} t_R(\vartheta) \quad \text{and} \quad \lim_{\vartheta \to \theta^+} t_L(\vartheta) \leq t_L(\theta).$$

(Note that the continuity of t_L and t_R in $(\theta - \varepsilon, \theta + \varepsilon)$ is guaranteed by the uniqueness of $\ell(\theta)$, therefore their right limits at θ are well defined.) Moreover, $t(\theta) = t_L(\theta)$; otherwise, $t(\theta) = t_R(\theta) \leq \lim_{\vartheta \to \theta^+} t_R(\vartheta)$ and thus p must have not crossed the bisecting line at θ.

There are two cases: $t(\theta + \varepsilon) = t_L(\theta + \varepsilon)$ or $t(\theta + \varepsilon) = t_R(\theta + \varepsilon)$ for sufficiently small $\varepsilon > 0$. For the former case, we have

$$t_R(\theta) \leq \lim_{\vartheta \to \theta^+} t_R(\vartheta) \leq \lim_{\vartheta \to \theta^+} t_L(\vartheta) \leq t_L(\theta).$$

If $t_L(\theta) = \lim_{\vartheta \to \theta^+} t_L(\vartheta)$, we are done. Otherwise, if $t_L(\theta) > \lim_{\vartheta \to \theta^+} t_L(\vartheta)$, then p should have belonged to $R(\theta)$, not to $L(\theta)$, a contradiction to the assumption; indeed we have $\lim_{\vartheta \to \theta^+} t_R(\vartheta) = |B_\theta(R(\theta) \cup \{p\})| \leq \lim_{\vartheta \to \theta^+} t_L(\vartheta) = |B_\theta(L(\theta) \setminus \{p\})| < t_L(\theta) = t(\theta)$. For the latter case, if $t_L(\theta) > \lim_{\vartheta \to \theta^+} t_R(\vartheta)$, we have $t_L(\theta) > \lim_{\vartheta \to \theta^+} t_R(\vartheta) = |B_\theta(R(\theta) \cup \{p\})| \geq \lim_{\vartheta \to \theta^+} t_L(\vartheta) = |B_\theta(L(\theta) \setminus \{p\})|$, a contradiction again as above. Hence, we have

$$t_R(\theta) \leq \lim_{\vartheta \to \theta^+} t_L(\vartheta) \leq t_L(\theta) \leq \lim_{\vartheta \to \theta^+} t_R(\vartheta).$$

If $t_L(\theta) = \lim_{\vartheta \to \theta^+} t_R(\vartheta)$, we are done. If $t_L(\theta) < \lim_{\vartheta \to \theta^+} t_R(\vartheta)$, then for sufficiently small $\varepsilon > 0$, $t(\theta + \varepsilon) = t_R(\theta + \varepsilon) > t_L(\theta) = t(\theta)$ and if p did not cross the bisecting line, we have a better solution, a contradiction to the optimality of $\ell(\theta + \varepsilon)$.

Therefore, at any $\theta \in [0, \pi)$ such that no two points in S lie on $\ell(\theta)$, t is continuous, implying that $\ell(\theta)$ moves continuously locally. Also, since we have at most $O(n^2)$ possibilities that two points in S lie on $\ell(\vartheta)$, this completes the proof of the lemma. $\square$

Now assume that a point is about to cross the bisecting line at orientation θ. Let $p_L \in L(\theta)$ be the last point lying on the right side of $B_\theta(L(\theta))$ in direction θ. Similarly, we let $p_R \in R(\theta)$ be the first point on the left side of $B_\theta(R(\theta))$ in direction θ. Since the bisecting line $\ell(\vartheta)$ moves continuously as ϑ increases from 0 to π, the crossing event occurs by p_L or p_R.

We consider the case that p_L crosses $\ell(\theta)$. The other case is symmetric. Consider the moment of the crossing of p_L: p_L lies on $\ell(\theta)$ and $B_L(\ell(\theta)) = B_\theta(L(\theta))$. Here, we have two possibilities; either another point of S also lies on $\ell(\theta)$ or not. If p_L is the only point on $\ell(\theta)$, the crossing event occurs because $|B_\theta(L(\theta))| \geq |B_\theta(R(\theta))|$ and $|B_{\theta+\varepsilon}(L(\theta))| > |B_{\theta+\varepsilon}(R(\theta) \cup \{p_L\})|$ for some arbitrarily small positive ε.

We characterize crossing events as follows. Let $f_L(\theta) := |B_\theta(L(\theta))|$ and $f_R(\theta) := |B_\theta(R(\theta))|$. Then, $f(\theta) = \max\{f_L(\theta), f_R(\theta)\}$. We also let $g_L(\theta)$ and $g_R(\theta)$ denote the areas of $B_\theta(L(\theta) \setminus \{p_L\})$ and $B_\theta(R(\theta) \cup \{p_L\})$, respectively, and let $g(\theta) := \max\{g_L(\theta), g_R(\theta)\}$.

Lemma 6 *If a crossing event occurs at θ and p_L crosses the bisecting line $\ell(\theta)$, either (1) two points of S, including p_L, lie on $\ell(\theta)$ or (2) $f(\theta) = g(\theta)$.*

Proof. Suppose that $S \cap \ell(\theta) = \{p_L\}$ but $f(\theta) \neq g(\theta)$. If $f(\theta) > g(\theta)$, then there always exists a small positive δ such that $f(\theta - \varepsilon) > g(\theta - \varepsilon)$ for any $0 < \varepsilon < \delta$ since $f(\theta)$ and $g(\theta)$ are continuous in every domain where no crossing event occurs. This contradicts the optimality of $\ell(\theta - \varepsilon)$. Now assume that $f(\theta) < g(\theta)$. Then again there always exists a small positive δ such that $\max\{|B_{\theta+\varepsilon}(L(\theta))|, |B_{\theta+\varepsilon}(R(\theta))|\} < \max\{|B_{\theta+\varepsilon}(L(\theta) \setminus \{p_L\})|, |B_{\theta+\varepsilon}(R(\theta) \cup \{p_L\})|\}$ for any $0 < \varepsilon < \delta$. This contradicts the optimality of $\ell(\theta + \varepsilon)$. ◻

The next crossing event of case (1) from the current θ can be predicted easily: we check if the line through p_L and p_R is parallel to some $\theta' \in [\theta, \theta_n)$ for the case where two points lying on the bisecting line are p_L and p_R. Otherwise, it occurs simultaneously with a non-crossing event so that we check if p_L crosses the bisecting line or not at each of such non-crossing events. Thus, the candidates of the next crossing event of case (1) can be computed in constant time at each loop of the algorithm.

For a crossing event of case (2), we monitor not only $f(\theta)$ but also $g(\theta)$ and check when they have the the same value after the current orientation θ. This can be done by solving $f(\vartheta) = g(\vartheta)$ in the interval $[\theta, \theta_n)$ where no non-crossing event occurs and then taking the smallest zero, at which we would have the next crossing event. Recall that $\ell(\vartheta)$ contains at most one point of S in the interval $[\theta, \theta_n)$. By Lemma 4, each of $f(\vartheta)$ and $g(\vartheta)$ has at most two breakpoints in the domain $[\theta, \theta_n)$ (where $f_L(\vartheta) = f_R(\vartheta)$ or $g_L(\vartheta) = g_R(\vartheta)$ holds) and the equation $f(\vartheta) = g(\vartheta)$ has at most a constant number of zeros (roughly at most 24.) The case when p_R crosses $\ell(\theta)$ can be handled with the functions h_L and h_R defined symmetrically to be $|B_\theta(L(\theta) \cup \{p_R\})|$ and $|B_\theta(R(\theta) \setminus \{p_R\})|$, and with $h(\theta) = \max\{h_L(\theta), h_R(\theta)\}$. Since the functions f, g, and h are all sinusoidal in the domain, we can compute all their crossings in constant time. These functions are determined once D_θ is fixed. Therefore, at each iteration of the algorithm, we can find the candidates of the next crossing event of case (2) in constant time. Note that it need not necessarily be true that we have a crossing event whenever $f(\theta) = g(\theta)$ or $f(\theta) = h(\theta)$. Thus, when we compute the next crossing event, we should test if it is a "real" crossing event; this can be done simply by checking the local behavior of the functions also in constant time.

To bound the total number of crossing events, we count the number of possible crossing events that occur on a certain point $p \in S$ during the rotation. For this, we need the following lemma. Let $\ell_p(\theta)$ denote the θ-oriented directed line through p.

Lemma 7 *The sequence of the four determinators of the θ-oriented bounding box of the points lying strictly in the left side of $\ell_p(\theta)$ changes at most $O(n)$ times while θ increases continuously from 0 to π.*

Proof. Let p_θ be the closest point to $\ell_p(\theta)$ among the points $L_{\ell_p(\theta)}$ lying in the left side of $\ell_p(\theta)$. We rotate $\ell_p(\theta)$ around p by increasing θ. Whenever a side of the bounding box $B_\theta(L_{\ell_p(\theta)})$, except the one determined by p_θ, touches two points p_1 and p_2 in S, this corresponds to a combinatorial change in the *quadrant hull* of S [2]; the quadrant hull is also known as the rectilinear convex hull, and there are only linear number of combinatorial changes to the quadrant hull while we rotate the point set from 0 to π. Observe that there always exists a quadrant defined by the line through p_1 and p_2, and a line orthogonal to the side touching p_1 and p_2 such that it contains p_1 and p_2 on one of its sides but contains no points of S in its interior. Therefore, there are at most $O(n)$ determinator changes in total to the three sides.

For a point q, we define a function $d_q(\theta)$ be the signed distance from q to $\ell_p(\theta)$, being positive when q lies to the left of $\ell_p(\theta)$. This function can be expressed as follows:

$$d_q(\theta) := |pq| \sin(\phi_q - \theta),$$

where ϕ_q is the orientation of the directed line from p to q.

We will look into the arrangement of the graphs of the functions d_q for all $q \in S, q \neq p$ in the domain $[0, \pi)$. Observe that p_θ changes at an orientation θ if one of the following conditions hold: (1) $d_q(\theta) = 0$ for any $q \in S \setminus \{p\}$, or (2) $d_q(\theta) = d_r(\theta) > 0$ for $q \neq r$ and $d_s(\theta) > d_q(\theta)$ for all $s \in L_{\ell_p(\theta)} \setminus \{q, r\}$. Consider now the graph of the function d_q for every $q \in S \setminus \{p\}$, and let C be the set of all the maximal pieces of the graphs lying above the x-axis. Then the number of changes of p_θ during the rotation is upper bounded by the combinatorial complexity of the lower envelope of the arrangement of C. Observe that $|C| \leq n - 1$.

To bound the complexity of the lower envelope, we let $C_1 \subseteq C$ be the set of curves intersecting the vertical line through the origin and $C_2 \subseteq C$ be the set of curves intersecting the vertical line through the point $(\pi - \varepsilon, 0)$ for sufficiently small positive ε. Observe that $C_1 \cup C_2 = C$ since each curve in C is defined by a graph of a sinusoidal function d_q of angular frequency 1. Moreover, since there is a vertical line intersecting all the curves in C_1, the lower envelope of C_1 has complexity linear in $|C_1|$ [12]. This also holds for the curves in C_2. Clearly, the lower envelope of C is the lower envelope of those of C_1 and C_2, and its combinatorial complexity is linear in $|C|$. Therefore, we conclude that the lower envelope of C has $O(n)$ complexity, and p_θ changes at most $O(n)$ times.

$\square$

Now we are ready to bound the number of crossing events.

Lemma 8 *The number of solutions to $f(\vartheta) = g(\vartheta)$ or $f(\vartheta) = h(\vartheta)$ for $\vartheta \in [0, \pi)$ is at most $O(n^2)$. Also, there are at most $O(n^2)$ crossing events.*

Proof. Recall Lemma 6. The number of crossing events falling in case (1) is bounded by $O(n^2)$, simply because there are at most $O(n^2)$ such distinct orientations. So from now on we assume that there is at most one point lying on the bisecting line, and count the number of the crossing events of case (2). We discuss about the case where $f(\theta) = g(\theta)$ only since the other case is symmetric. Note that $f(\theta) = g(\theta)$ if and only if $f_L(\theta) = g_R(\theta)$.

Let $f_L^p(\theta) := |B_\theta(L_{\ell_p(\theta)})|$ and $g_R^p(\theta) := |B_\theta(R_{\ell_p(\theta)})|$, where p belongs to $L_{\ell_p(\theta)}$. In orientation θ, if p is the rightmost point of $L(\theta)$, then $f_L^p(\theta) = f_L(\theta)$ and $g_R^p(\theta) = g_R(\theta)$. Hence, in this proof, we rotate $\ell_p(\theta)$ by increasing θ and count the number of times when $f_L^p(\theta) = g_R^p(\theta)$ for all $p \in S$, which bounds from above the possible number of the orientations θ such that $f(\theta) = g(\theta)$.

Consider an orientation interval (α, β) where we have no change in the 8 determinators on the sides of the two bounding boxes. In such an interval, $f_L^p(\theta)$ and $g_R^p(\theta)$ are sinusoidal functions by Lemma 2, and their graphs intersect at most twice by Lemma 4. Therefore, in (α, β), there are at most two such θ that $f_L^p(\theta) = g_R^p(\theta)$.

Thus, the only thing left is to bound the number of such intervals (α, β). Lemma 7 tells us that the sequence of the four determinators of $B_\theta(R_{\ell_p(\theta)})$ changes at most $O(n)$ times. For $B_\theta(L_{\ell_p(\theta)})$, we have one fixed determinator p; thus the number of changes in its four determinators is also bounded by $O(n)$. For each point $p \in S$, we have $O(n)$ such intervals (α, β), and thus the first statement is shown.

Also, whenever a crossing event of case (2) occurs, a point in S lies on the bisecting line $\ell(\theta)$ and we have $f(\theta) = g(\theta)$ or $f(\theta) = h(\theta)$ by Lemma 6. Thus, the number of crossing events is bounded by $O(n^2)$. ◻

Consequently, we spend at most $O(n^2)$ time to compute all events in total, and thus we repeat the main loop $O(n^2)$ times while each run of the main loop takes $O(\log n)$ amortized time. Finally, we conclude the following.

Theorem 1 *Given a set S of n points, an optimal pair of two disjoint and parallel rectangles containing all points in S can be computed in $O(n^2 \log n)$ worst-case time and $O(n)$ space.*

Tight example construction Here, we describe how to construct a problem instance which yields at least $\Omega(n^2)$ number of events; thus, our upper bound is asymptotically tight. For any positive integer $n > 3$, let $m = 2\lfloor \frac{n}{4} \rfloor + 1$. Then, m is an odd number, $m = \Theta(n)$, and $n - m = \Theta(n)$. Now, place m points at the corners of a regular m-gon bounded by a unit circle centered at the origin, with one point placed at coordinate $(0, 1)$. The remaining $n - m$ points are placed in a disk U centered at the origin with radius $\varepsilon > 0$, where ε is sufficiently small positive number. Let S_n be this constructed set of points.

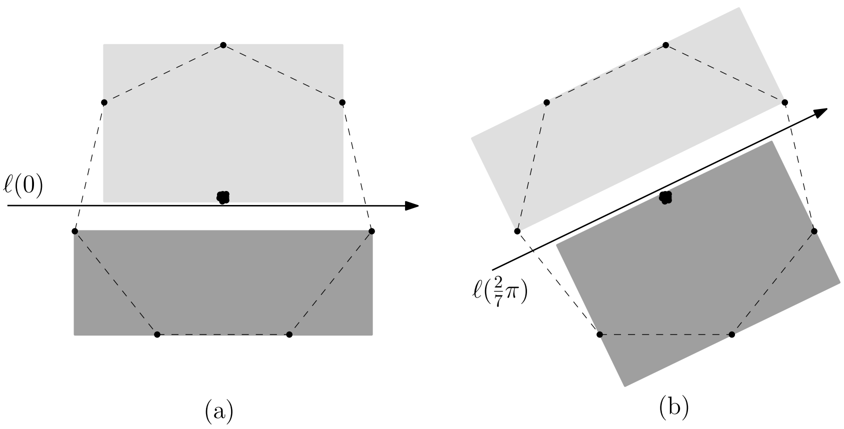

Figure 4: Tight example construction when $n = 13$ and $m = 7$. The optimal two-rectangle covering in orientation (a) 0 and (b) $2\pi/m = \frac{2}{7}\pi$. The light gray rectangle is $B_\theta(L(\theta))$ and the dark gray rectangle is $B_\theta(R(\theta))$

The optimal two-rectangle coverings of $S = S_n$ in orientation θ are as follows: If ε is small enough, one of $L(0)$ and $R(0)$ includes all the points in $S \cap U$. Without loss of generality, suppose that $L(0)$ includes $S \cap U$ as shown in Figure 4(a). On the other hand, at orientation $2\pi/m$, $R(2\pi/m)$ consists of the points lying in U; this is easy to see since our point set is symmetric except for points lying in U (see Figure 4(b)). Indeed, $L(2i\pi/m)$ includes the points lying in U but $L((2i+1)\pi/m)$ does not, for $i = 0, \ldots, \lfloor m/2 \rfloor$. Thus, we have at least $n - m$ crossing events per each π/m rotation of orientation, which implies that we have at least $\Omega(n^2)$ number of events for the instance S_n.

3 Covering Points by Two Disjoint Non-Parallel Rectangles

In this section, we consider the two-rectangle covering problem where two bounding boxes are not necessarily parallel. More specifically, we find an optimal pair of two disjoint rectangles containing the given set S of points such that one rectangle is axis-parallel and the other rectangle is not necessarily axis-parallel. Informally speaking, one rectangle is free to rotate and we call it the *free* rectangle. We further consider the variant of the problem where both rectangles are free.

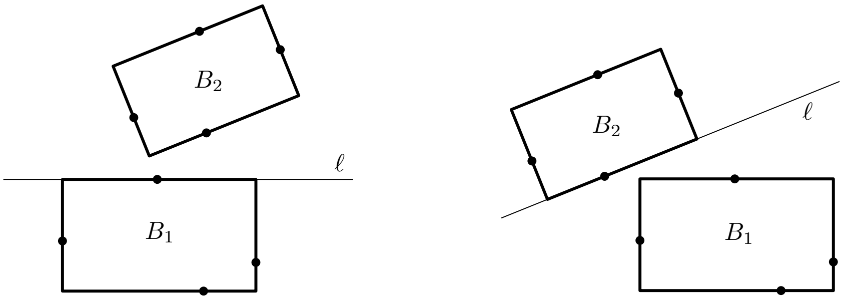

Figure 5: There always exists a separating line ℓ that supports a side of the axis-parallel rectangle B_1 (left) or a side of the free rectangle B_2 (right).

Consider an optimal solution for S, consisting of an axis-parallel rectangle B_1 and a θ-oriented rectangle B_2 for $0 \leq \theta < \pi/2$. We observe that there always exists a line ℓ separating B_1 and B_2 which supports a side of B_1 or a side of B_2. We have two possibilities as shown in Figure 5:

1. ℓ supports a side of B_1, therefore, is either horizontal or vertical, or

2. ℓ supports a side of B_2.

Our algorithm, to be described below, simply seeks an optimal two-rectangle covering for S in each case above, from which we find the optimal one. The following theorem summarizes the result.

Theorem 2 *Let S be a set of n points in the plane. One can compute in $O(n^2 \log n)$ time with $O(n)$ space an optimal pair of two disjoint rectangles covering all points in S such that one is axis-parallel and the other is in arbitrary orientation.*

Remind that $B_\theta(S)$ is the θ-oriented bounding box of the set S.

Case 1 We assume without loss of generality that ℓ is horizontal and it supports the top side of the axis-parallel rectangle. The other cases can be handled in a symmetric way. Since the top side contains a point in S, we have only n candidates for ℓ. Thus, we are done by computing the possible minimum free rectangle above the horizontal line ℓ through each $p \in S$.

Let $p_1, \ldots, p_n$ be the list of the points in S sorted in the increasing order of their y-coordinates, and ℓ_i be the horizontal line through p_i. Let $S_1(i) := \{p_1, \ldots, p_i\}$ be the subset of S consisting of the points lying on or below ℓ_i and $S_2(i) := S \setminus S_1(i)$. Note that there can be two points lying on ℓ_i, and we consider both of them to be in $S_1(i)$: if one of them does not lie on the top side of $B_0(S_1(i))$, it must be the bottommost corner of the other rectangle and there exists a separating line supporting $B_\phi(S_2(i))$, which falls in case (2).

We let $f_1(i)$ to be the area of $B_0(S_1(i))$ and

$$f_2(i) := \min_{\phi: B_\phi(S_2(i)) \text{ is disjoint from } \ell_i} |B_\phi(S_2(i))|.$$

We then seek a point $p_i \in S$ such that $\max\{f_1(i), f_2(i)\}$ is minimized for all $1 \le i \le n$. Hence, the most difficult part of the algorithm is to evaluate $f_2(i)$ in this case.

To evaluate $f_2(i)$, we compute the range Φ_i of ϕ where $B_\phi(S_2(i))$ is disjoint from ℓ_i. Since $S_2(i)$ is fixed, we can compute the description of $|B_\varphi(S_2(i))|$ as a function of $\varphi \in [0, \pi/2)$ in $O(n \log n)$ time by the rotating caliper technique [14]. If $B_\phi(S_2(i))$ and ℓ_i are not disjoint, then one corner of the rectangle lies below ℓ_i. Thus, we compute the locus of the corners of $B_\varphi(S_2(i))$ as φ increases from 0 to $\pi/2$. This locus is a simple closed curve consisting of $O(n)$ circular arcs; this curve is known as the *angle hull* $\mathcal{AH} := \mathcal{AH}(\mathcal{CH}(S_2(i)))$ of the convex hull $\mathcal{CH}(S_2(i))$ of $S_2(i)$, defined to be the locus of points x such that the two tangent lines to $\mathcal{CH}(S_2(i))$ through x make the right angle [6]. See Figure 6.

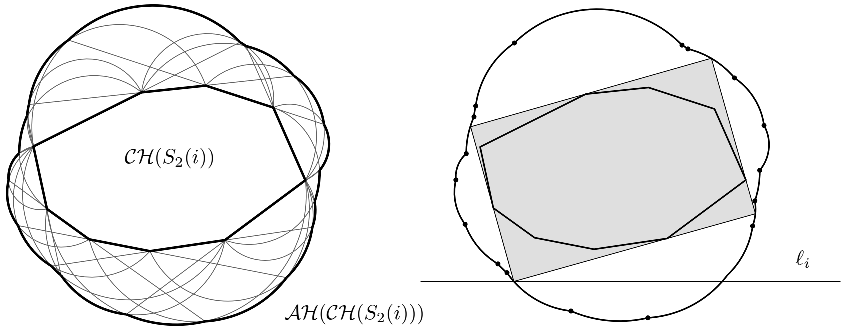

Figure 6: The angle hull $\mathcal{AH}(\mathcal{CH}(S_2(i)))$ (left), and the bounding box of $S_2(i)$ at an intersection of ℓ_i with the angle hull (right).

Each point x on $\mathcal{AH}$ is mapped to an orientation $\phi \in [0, \pi/2)$ of the bounding box of $S_2(i)$ one of whose corners lies at point x. Moreover, each endpoint of an arc of $\mathcal{AH}$ corresponds to a breakpoint of the function $|B_\varphi(S_2(i))|$, that is, an orientation ϕ where two points of $S_2(i)$ lie on a side of $B_\phi(S_2(i))$. The following lemma follows directly from earlier results [14].

Lemma 9 *The value of $f_2(i)$ is realized as $|B_\phi(S_2(i))|$ such that $\phi \in \Phi_i$, and either (1) two points of $S_2(i)$ lie on one side of $B_\phi(S_2(i))$ or (2) one corner of $B_\phi(S_2(i))$ lies on ℓ_i.*

The first case in the above lemma can be handled by checking each breakpoint of $|B_\varphi(S_2(i))|$; construct $B_\phi(S_2(i))$ for each such breakpoint ϕ and check whether it intersects ℓ_i or not. This takes $O(1)$ time per each breakpoint. For the second case, we compute the intersection of ℓ_i and $\mathcal{AH}$ in $O(n)$ time and take the minimum value among $|B_\phi(S_2(i))|$ for ϕ corresponding to each intersection point between ℓ_i and $\mathcal{AH}$. Then, $f_2(i)$ is the minimum value among those computed as above.

All this process for evaluating $f_2(i)$ takes only $O(n)$ time once the convex hull of $S_2(i)$ is computed. Hence, for each i, we can compute in $O(n \log n)$ time the convex hull of $S_2(i)$, the description of function $|B_\phi(S_2(i))|$, the angle hull $\mathcal{AH}$, the intersection of ℓ_i and $\mathcal{AH}$, and the value of $f_2(i)$.

Lemma 10 *An optimal solution for case 1 can be found in $O(n^2 \log n)$ time with $O(n)$ space.*

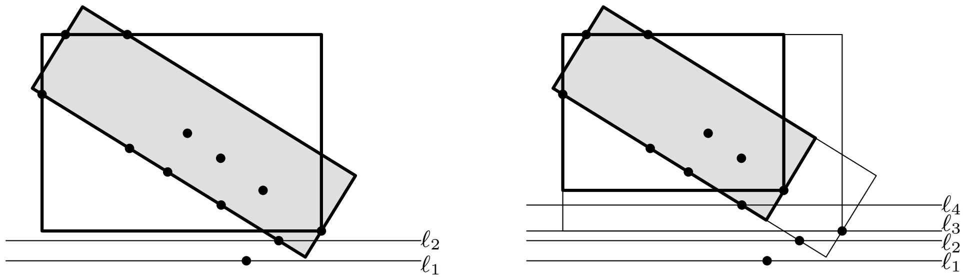

Figure 7: An example of points S where $f_2(i)$ is not monotone. Observe that $f_2(3) < f_2(1) < f_2(4) < f_2(2)$.

Remarks : One might be curious about whether $f_2(i)$ is monotone or not; if it were monotone, one could apply a binary search on i to get a better performance. However, it is not necessarily true by a simple example. See Figure 7.

The running time of the algorithm can be improved to $O(n^2)$ by using the dynamic convex hull structure by Brodal and Jacob [4]. But this is not very meaningful since our algorithm for the other case takes $O(n^2 \log n)$ time anyway.

Case 2 In this case, we use a bit different approach. Consider a bipartition L and R of S by a partitioning line ℓ, where R is the set of points lying to the right of ℓ and $L = S \setminus R$. Then, the set of valid orientations for ℓ to get the bipartition is expressed as an orientation interval $(\alpha, \beta) \subset [0, 2\pi)$ (after a proper rotation of the whole S). An optimal solution with an axis-parallel rectangle $B_0(R)$ and a free rectangle $B_\theta(L)$ for $\theta \in (\alpha, \beta)$ can be found by computing $B_0(R)$ and finding the minimum $B_\theta(L)$ for $\theta \in (\alpha, \beta)$ such that both are disjoint from each other.

More formally, we let $\ell_p(\theta)$ be the θ-oriented directed line passing through $p \in S$, and let $L_p(\theta)$ and $R_p(\theta)$ be the bipartition of S by $\ell_p(\theta)$ where $p \in L_p(\theta)$. Let $f_1^p(\theta) := |B_0(R_p(\theta))|$ and $f_2^p(\theta) := |B_\theta(L_p(\theta))|$. We rotate $\ell_p(\theta)$ by increasing θ from 0 to 2π and gather local minima of $\max\{f_1^p(\theta), f_2^p(\theta)\}$. By Lemmas 2, 4, and 7, for fixed $p \in S$, $f_2^p(\theta)$ has $O(n)$ breakpoints and is divided into the same asymptotic number of sinusoidal pieces, in each of which we have no change of the determinators of $B_\theta(L_p(\theta))$ and further those of $B_0(R_p(\theta))$. Here, we redefine three types of events to be handled.

- *Non-crossing event*: When the determinators of $B_\theta(L_p(\theta))$ changes while $L_p(\theta)$ remains the same.
- *Crossing event*: When $L_p(\theta)$ changes, that is, another point $q \in S$ lies on $\ell_p(\theta)$.
- *Rectangle-touching event*: When $B_0(R_p(\theta))$ touches $\ell_p(\theta)$, that is, $\ell_p(\theta)$ is tangent to $B_0(R_p(\theta))$.

Our algorithm for this case is described as follows:

Algorithm *OneAxisParallelAndOneFreeRectangleCoverCase2(S)*
($*$ computes the optimal two rectangles in Case 2 $*$)
1. **for** each $p \in S$
2. **do** initialize a dynamic convex hull $\mathcal{CH}$ and an event queue $\mathcal{Q}$
3. compute $L_p(0)$ and add the points in $L_p(0)$ into $\mathcal{CH}$, and compute $R_p(0)$ and $B_0(R_p(0))$

4. $\theta \leftarrow 0$
5. compute all crossing events for p and put them into $\mathcal{Q}$
6. compute all non-crossing and rectangle-touching events before the first crossing event and put them into $\mathcal{Q}$
7. **while** $\theta < 2\pi$
8. **do** pop the next upcoming event at θ' from $\mathcal{Q}$
9. **if** for any $\vartheta \in (\theta, \theta')$, $B_0(R_p(\vartheta))$ dose not intersect ℓ_p
10. **then** minimize $\max\{f_1^p(\vartheta), f_2^p(\vartheta)\}$ over $\vartheta \in [\theta, \theta')$
11. **if** the event at θ' is a crossing event
12. **then** update $L_p(\theta')$, $R_p(\theta')$, $\mathcal{CH}$, and $B_0(R_p(\theta'))$
13. compute all non-crossing and rectangle-touching events between θ' and the next crossing event, and put them into $\mathcal{Q}$
14. $\theta \leftarrow \theta'$
15. **return** the minimum objective value with its orientation

Crossing events occur when $q \in S$ with $q \neq p$ lies on $\ell_p(\theta)$; thus, they can be computed before the **while** loop. Since $L_p(\theta)$ and $R_p(\theta)$ do not change between two consecutive crossing events, we can compute all non-crossing and rectangle-touching events in time proportional to the number of the events with $\mathcal{CH}$ and $B_0(R_p(\theta))$. By Lemma 7, we know that the number of possible non-crossing events is $O(n)$ for each $p \in S$. The number of rectangle-touching events can be bound by the number of crossing events; the number of lines through p which are tangent to a rectangle is at most two unless p is a corner of the rectangle.

Lemma 11 *The total number of events while θ increases from 0 to 2π is at most $O(n)$ for each $p \in S$.*

Finally, we conclude the following.

Lemma 12 *For each point $p \in S$, the algorithm above computes all the local minima of $\max\{f_1^p(\theta), f_2^p(\theta)\}$ in $O(n \log n)$ time with $O(n)$ space for $\theta \in [0, 2\pi)$. Thus, an optimal solution of case 2 can be found in $O(n^2 \log n)$ time and $O(n)$ space.*

Proof. For each $p \in S$, the initialization step (Lines 2–5) takes $O(n \log n)$ time. Each iteration of the **while** loop takes $O(\log n)$ time: computing non-crossing and rectangle-touching events takes $O(1)$ time per each. Each operation on the event queue $\mathcal{Q}$ takes $O(\log n)$ time by using a standard priority queue. Computing local minima in the domain $[\theta, \theta')$ takes $O(1)$ time by Lemma 4 since $f_1^p(\vartheta)$ is constant and $f_2^p(\vartheta)$ is sinusoidal with angular frequency 2 in $[\theta, \theta')$ for fixed $p \in S$ by Lemma 2. For each crossing event, only one point is moved from one side to the other side; thus, it takes $O(\log n)$ time to update the data structures and the invariants of the algorithm. Finally, Lemma 11 states that we have at most $O(n)$ events and hence that we run the main loop $O(n)$ times. ◻

3.1 Some remarks about two free rectangles

We conclude this paper with discussion about the case of two disjoint free rectangles. To express the orientations of two such rectangles, we use two symbols θ and ϕ for their orientations. As observed above, for any optimal disjoint two free rectangles (B_1, B_2) for S, there exists a line ℓ that separates B_1 and B_2 and supports one side of the two rectangles. We thus can search only those rectangles such that B_1 is θ-oriented, B_2 is ϕ-oriented, B_1 and B_2 are disjoint, and a θ-oriented line ℓ supporting a side of B_1 separates B_1 and B_2 for each $\theta, \phi \in [0, 2\pi)$.

For each $p \in S$, we let $\ell_p(\theta)$ be the directed line through p in direction θ and $L_p(\theta)$ and $R_p(\theta)$ be defined as above. Also, let $f_1^p(\theta) := |B_\theta(L_p(\theta))|$ and $f_2^p(\theta) := \min_{\phi \in \Phi_p(\theta)} |B_\phi(R_p(\theta))|$, where $\Phi_p(\theta)$ is the set of orientations ϕ such that $B_\phi(R_p(\theta))$ is disjoint from $\ell_p(\theta)$. As in Case 1 of the above, there always exists $\phi \in \Phi_p(\theta)$ such that $f_2^p(\theta) = |B_\phi(R_p(\theta))|$ and either (1) a corner of $B_\phi(R_p(\theta))$ lies on $\ell_p(\theta)$ or (2) two points in $R_p(\theta)$ lies on a side of $B_\phi(R_p(\theta))$. Note that for fixed θ we can compute each orientation ϕ where a corner of $B_\phi(R_p(\theta))$ lies on $\ell_p(\theta)$ by intersecting the angle hull $\mathcal{AH}$ and $\ell_p(\theta)$. The difficulty here is that we should search every $\theta \in [0, 2\pi)$, and thus each such ϕ is indeed expressed as a function $\phi(\theta)$ of θ. Unfortunately, such a function $\phi(\theta)$ is very complicated; it appears as the inverse of the cosine of a function of $\sin\theta$ and $\cos\theta$ containing a square root term:

$$\arccos\left(\sin\theta\,(c_1 \sin\theta) \pm \cos\theta\sqrt{1 - (c_1 \sin\theta)^2}\right) + c_2,$$

where c_1 and c_2 are constants. It seems very difficult to analyze the functions of this form and to use the same approach as in the previous variations of the problem.

References

[1] P. Agarwal and M. Sharir. Efficient algorithms for geometric optimization. *ACM Comput. Surveys*, 30:412–458, 1998.

[2] S. W. Bae, C. Lee, H.-K. Ahn, S. Choi, and K.-Y. Chwa. Maintaining extremal points and its applications to deciding optimal orientations. In *Proc. 18th Annu. Internat. Sympos. Algo. Comput. (ISAAC)*, volume 4835 of *LNCS*, pages 788–799, 2007.

[3] S. Bespamyatnikh and M. Segal. Covering a set of points by two axis-parallel boxes. *Inform. Process. Lett.*, 75:95–100, 2000.

[4] G. S. Brodal and R. Jacob. Dynamic planar convex hull. In *Proc. 43rd Annu. IEEE Sympos. Found. Comput. Sci. (FOCS)*, pages 617–626, 2002.

[5] G. Frederickson and D. Johnson. Generalized selection and ranking: sorted matrices. *SIAM Journal on Computing*, 13:14–30, 1984.

[6] F. Hoffmann, C. Icking, R. Klein, and K. Kriegel. The polygon exploration problem. *SIAM J. Comput.*, 31(2):577–600, 2001.

[7] J. W. Jaromczyk and M. Kowaluk. Orientation indenpendent covering of point sets in r^2 with pairs of rectangles or optimal squares. In *Proc. Europ. Workshop Comput. Geom. (EuroCG)*, pages 54–61, 1996.

[8] M. Kats, K. Kedem, and M. Segal. Discrete rectilinear 2-center problem. *Comput. Geom.: Theory and Appl.*, 15:203–214, 2000.

[9] M. Katz and M. Sharir. An expander-based approach to geometric optimization. *SIAM J. Comput.*, 26:1384–1408, 1997.

[10] N. Megiddo. Applying parallel computation algorithms in the design of serial algorithms. *J. ACM*, 30:852–865, 1983.

[11] C. Saha and S. Das. Covering a set of points in a plane using two parallel rectangles. In *Proc. of Internat. Conference on Comput.: Theory and Appl. (ICCTA)*, pages 214–218, 2007.

[12] M. Sarir and P. Agarwal. *Davenport-Schinzel Sequences and Their Geometric Applications*. Cambridge University Press, New York, 1995.

[13] M. Segal. *Covering point sets and accompanying problems*. PhD thesis, Ben-Gurion University, Israel, 1999.

[14] G. Toussaint. Solving geometric problems with the rotating calipers. In *Proc. of IEEE MELECON'83*, Athens, Greece, 1983.