# Covering a Set of Points in a Plane using Two Parallel Rectangles

## Chandan Saha

*Indian Institute of Technology Kanpur, Kanpur-208016, India*

## Sandip Das

*Indian Statistical Institute, Kolkata-700108, India*

**Abstract**

We consider the problem of finding two parallel rectangles, in arbitrary orientation, covering a given set of $n$ points in a plane, such that the area of the larger rectangle is minimized. We give a simple algorithm that solves the problem in $O(n^3)$ time using $O(n^2)$ space. Without altering the complexity, the algorithm can be modified to solve another optimization problem, namely, minimize the sum of the areas of two arbitrarily oriented parallel rectangles covering a given set of points in a plane.

*Key words:* Algorithms, Computational Geometry, Covering, Optimization, Rectangles.

## 1 Introduction

Two rectangles are said to be *parallel* if each side of a rectangle is parallel to a side of the other rectangle. In this paper we consider the following problem **P**: Given a set $S$ of $n$ points in a plane, locate two parallel rectangles $D_1$ and $D_2$ that cover $S$ in such a way that the area of the larger rectangle is minimized, among all possible covers by two parallel rectangles. Figure 1 depicts a possible location of two parallel rectangles covering point set $S$ in a plane.

Bespamyatnikh et al. (1997) (1) considered a similar problem in $d$-dimensional space, but using *axis-parallel* boxes to cover the points, and obtained a time complexity of $O(n \log n + n^{d-1})$. In 1996, Jaromczyk et al. (2) proposed an
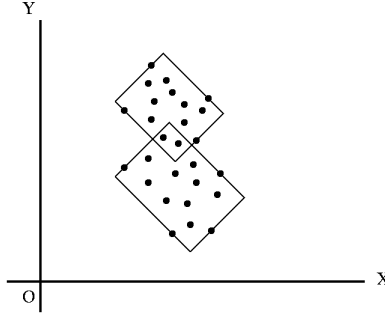
---

Fig. 1. Covering points by two parallel rectangles.

$O(n^2)$ time algorithm that finds optimal covering of $S$ using two squares with mutually parallel sides and optimizes the sizes of the squares with respect to their side lengths. This was followed by a work by Katz, Kedem and Segal (3) in 1999 that solved the problem of finding two constrained parallel squares, covering $S$, so as to minimize the size of the larger square. The running time of their algorithm is $O(n^2 \log^4 n)$ and uses $O(n^2)$ space. In the same paper, they also presented an $O(n^3 \log^2 n)$ time algorithm using $O(n^2)$ space to locate the optimum squares, where each square is allowed to rotate independently.

In this paper, we propose an efficient algorithm that solves problem **P** in $O(n^3)$ time using $O(n^2)$ space. The result compares well with (2) and (3), as a rectangle has an added degree of freedom compared to that of a square. Problem **P** finds application in VLSI physical design for accommodating specified locations like hot spots, power pins into two parallel rectangles. It also has application in the field of geographical information systems for map data modeling, map overlay, map labeling etc. Moreover, there are applications in image processing and facility location problems.

## 2 Preliminaries

Let $S = \{p_1, p_2, \ldots, p_n\}$ be the set of $n$ points. $x(p)$ and $y(p)$ represent the $x$ and $y$ coordinates of point $p$, respectively. $XY(\alpha, p)$ is the coordinate system obtained by making a counter-clockwise rotation of the original axis by an angle $\alpha \in [0, \pi/2)$ around origin $O$ and by shifting the origin from $O$ to point $p$ using linear translation. We denote $XY(\alpha, O)$ by $XY(\alpha)$. For $\alpha \in [0, \pi/2)$, $\wp(\alpha)$ is the subset of $S$ with $p \in \wp(\alpha)$ if and only if there exists at least one *empty* closed quadrant of the system $XY(\alpha, p)$ (see figure 2). By *empty* closed quadrant we mean that no point is present in the quadrant except $p$.

Let $D_1$ and $D_2$ be two optimal parallel rectangles covering sets of points $S_1$ and $S_2$, respectively, where $S_1 \cup S_2 = S$. If $Area(D_1) \geq Area(D_2)$ then $D_1$ and $D_2$ are the smallest axis-aligned rectangles covering $S_1$ and $S_2 \backslash S_1$ respectively, with respect to some system $XY(\alpha)$. Let $D(\alpha)$ be the smallest axis-aligned rectangle in $XY(\alpha)$ that encloses all points of $S$. $T(\alpha)$ is an
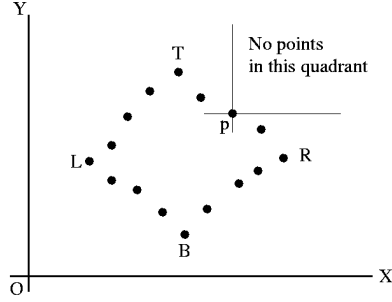
Fig. 2. Points in $\wp(0)$.

ordered set containing the top, left, bottom and right boundary points of $D(\alpha)$ in that order. Call the set $T(\alpha)$ the *extreme points* of $S$ in the system $XY(\alpha)$. $XY$ is the coordinate system $XY(\alpha)$ at $\alpha = 0$. For simplicity, assume that no two points in $S$ have the same $x$ or $y$ coordinate in $XY$. Also assume that no three points are collinear and no three points form an angle of $\frac{\pi}{2}$. These cases can be handled with minor modifications to our algorithm.

## 3  Algorithm for axis-parallel rectangles

In this section, we review the case where the optimum rectangles $D_1$ and $D_2$ are parallel to the axes of $XY$ (also known as *isothetic* rectangles).

### 3.1  The rectangles do not overlap

In this case, there exists a vertical or a horizontal line that partitions the set $S$ into two subsets $S_1$ and $S_2$. If two sorted sequences of points, $\chi$ and $\psi$ (ordered with respect to the $x$ and $y$ coordinates), are available then we can use a sweep line algorithm to locate the optimal rectangles in $O(n)$ time.

### 3.2  The rectangles may overlap

Assume that all the points on a boundary of one of the rectangle (say, $D_1$) are not contained within the other rectangle (say, $D_2$).

**Lemma 3.1** *If the rectangles $D_1$ and $D_2$ overlap, then each side of the rectangles must contain a point in $\wp(0)$. Moreover, if the sorted sequences $\chi$ and $\psi$ are available then the set $\wp(0)$ can be computed in $O(n)$ time.*

*Proof*: The first statement follows from the optimal nature of $D_1$ and $D_2$. Set $\wp(0)$ can be computed in $O(n)$ time using a sweep line algorithm.  ∎

Thus we can ignore all the points in $S$ except the elements in $\wp(0)$. From Lemma 3.1 and the results by Bespamyatnikh and Segal (1) we conclude,

**Theorem 3.1** *Given a set $S$ of $n$ points in a plane and the sorted sequences $\chi$ and $\psi$, the problem of locating two axis-parallel rectangles to cover $S$, so as to minimize the area of the larger rectangle, can be solved in $O(n)$ time.*

3

## 4    Covering by orientation independent parallel rectangles

Consider the case where the parallel rectangles may be placed in any orientation and form an angle in the range $[0, 2\pi]$ with $x$-axis in $XY$. With the intent of pruning our search space, we explore some characterizations of the optimal rectangles.

Let $CH(S)$ be the convex hull of point set $S$ and $\mathcal{A}(S)$ be the set of angles formed by the edges of $CH(S)$ with the $x$-axis. In case of covering points with a single rectangle, Freeman and Shapira (5) suggested an interesting characterization.

**Result 1** *(5) Let $D(\alpha)$ be the minimum area rectangle enclosing points of $S$ with one of its sides making an angle $\alpha \in [0, \frac{\pi}{2})$ with the $x$ axis of the $XY$ coordinate system. If the area of $D(\alpha)$ achieves a local optimal value then either $\alpha$ or $\alpha + \frac{\pi}{2}$ coincide with one of the elements in $\mathcal{A}(S)$.*

**Lemma 4.1** *Let one of the sides of rectangles $D_1$ and $D_2$ be inclined at angle $\alpha$ to the $x$-axis in $XY$ plane. Then at least one of the following two conditions hold:*

*(i)   there exists at least a pair of points $p$, $q$ in $S$ such that $x(p) = x(q)$ or $y(p) = y(q)$ with respect to the coordinate system $XY(\alpha)$;*
*(ii)   $Area(D_1) = Area(D_2)$.*

*Proof*: Suppose that, none of the conditions hold. Let $S_1$ and $S_2$ be the subsets of points in $S$ that are covered by $D_1$ and $D_2$ with $Area(D_1) > Area(D_2)$. Since condition (i) is not satisfied, there exist infinitesimal rotations of the two rectangles in both clockwise and counter-clockwise directions such that the rectangles still cover the same subsets $S_1$ and $S_2$, while their boundary points remain unaltered. From result 1 it follows that, there exists a small angle of rotation $\delta\alpha$ of the system $XY(\alpha)$ to $XY(\alpha + \delta\alpha)$ such that the area of rectangle $D_1$ is reduced while $D_1$ remains the larger of the two rectangles. This is contrary to the assumption that $D_1$ and $D_2$ forms an optimal pair of rectangles. ∎

The coordinate system $XY(\alpha)$, for $\alpha$ ($\in [0, 2\pi]$), is called *separable* if no two points $p$, $q$ in $S$ have same $x$ or $y$ coordinates in $XY(\alpha)$. An open interval $(a, b)$, within the range $[0, 2\pi]$, is called a *separable interval* if for any $\alpha \in (a, b)$, the system $XY(\alpha)$ is separable (see figure 3).

Let $\chi(\alpha)$ and $\psi(\alpha)$ be two sequences of points of $S$ sorted with respect to their $x$ and $y$-coordinates, respectively, in the coordinate system $XY(\alpha)$.

**Observation 1** *The sequences $\chi(\alpha)$ and $\psi(\alpha)$ is the same for any $\alpha \in \Delta$, where $\Delta$ is a separable interval.*
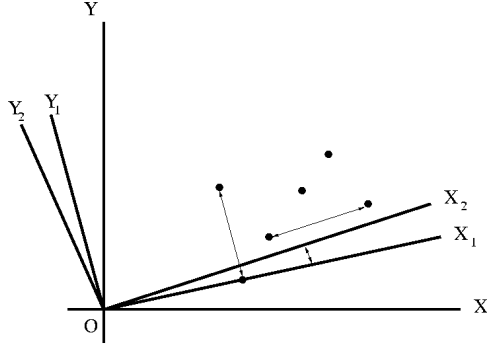
Fig. 3. The interval $XOX_1$ to $XOX_2$ is a separable interval.

Let $D(\alpha, S')$ be the smallest axis aligned rectangle in $XY(\alpha)$ enclosing the points of a set $S' \subseteq S$. Points on the boundary of the rectangle $D(\alpha, S')$ is termed as the *extreme points* of $D(\alpha, S')$.

**Observation 2** *Consider any separable interval $\Delta$ and a set $S' \subseteq S$. Rectangles $D(\alpha, S')$, for all $\alpha \in \Delta$, have the same set of extreme points.*

$S_1$ and $S_2$ be subsets of points in $S$ with $S_1 \cup S_2 = S$ ($S_1$ and $S_2$ need not be disjoint). $\Lambda_1$ and $\Lambda_2$ be the extreme points of $D(\alpha, S_1)$ and $D(\alpha, S_2)$, respectively, for $\alpha \in \Delta$.

**Lemma 4.2** *Given a separable interval $\Delta$, and the sets $\Lambda_1$, $\Lambda_2$, if there exist an angle $\alpha \in \Delta$ such that $Area(D(\alpha, S_1)) = Area(D(\alpha, S_2))$ then $\alpha$ can be computed in constant time. There can be at most two different solutions for $\alpha$ in any separable interval $\Delta$.*

*Proof:* Let $\Delta : (\phi_i, \phi_f)$ be a *separable interval*. The areas of rectangles $D(\alpha, S_1)$ and $D(\alpha, S_2)$ can be expressed as a trigonometric function of $\alpha$. Equating $Area(D(\alpha, S_1))$ and $Area(D(\alpha, S_2))$ we get an equation of the form, $F(\alpha) = C$, where $F(\alpha)$ is a sinusoidal function and $C$ is a constant. As $\alpha$ varies in the range $[0, \frac{\pi}{2}]$, the equation $F(\alpha) = C$ can have at most two solutions. ∎

Suppose $\Theta$ be the set of angles in $[0, \pi]$ formed by the lines, joining every pairs of points in $S$, with the $x$-axis of the $XY$ coordinate system. Note that, for any $\theta \in \Theta$, $0 \leq \theta \leq \pi/2$, the systems $XY(\theta)$ and $XY(\pi/2 + \theta)$ are *not* separable, and for $\pi/2 \leq \theta \leq \pi$, the systems $XY(\theta)$ and $XY(\theta - \pi/2)$ are *not* separable. Observe that, for a *non-separable* coordinate system $XY(\alpha)$ with $0 \leq \alpha \leq \pi/2$, either $\alpha$ or $\alpha + \pi/2$ must be an element of $\Theta$. Consider the set $\Phi = \{\phi : \phi \in [0, \pi/2], \phi \in \Theta \text{ or } (\phi + \pi/2) \in \Theta\}$. The cardinality of $\Phi$ is bounded by $\binom{n}{2}$. The elements, $\phi_1, \phi_2, \ldots, \phi_k$ of set $\Phi$ be in increasing order of their values and this ordered sequence is referred by $\Phi$ itself. As no two points have same $x$ or $y$ coordinates in $XY$, $\phi_1 > 0$. We introduce a new element $\phi_0 = 0$ and therefore, the set of intervals $\Delta = \{\Delta_1, \Delta_2, \ldots, \Delta_k\}$ where $\Delta_i = (\phi_{i-1}, \phi_i)$ for $i = 1, 2, \ldots, k$ are separable intervals in $[0, \pi/2]$. We introduce another

separable interval $\Delta_{k+1} = (\phi_k, \pi/2]$ and hence $\Delta = \{\Delta_1, \Delta_2, \ldots, \Delta_k, \Delta_{k+1}\}$ is the exhaustive set of separable intervals in $[0, \pi/2]$. Suppose, points $p_i$ and $q_i$ of $S$ have the same $x$ or $y$-coordinate values in the coordinate system $XY(\phi_i)$ ($\phi_i \in \Phi$). Given the sequences $\chi(\phi_i)$ and $\psi(\phi_i)$, if the indices of $p_i$ and $q_i$ in these sequences are available then for any $\alpha$ in $\Delta_i$ or $\Delta_{i+1}$ the sequences $\chi(\alpha)$ and $\psi(\alpha)$ can be evaluated in constant time by making constant number of interchanges in $\chi(\phi_i)$ and $\psi(\phi_i)$.

### 4.1 The rectangles do not overlap

We state the algorithm below.

**Optimal_disjoint_rectangles($S$)**

(1) Compute $\Theta$ and enumerate the set $\Phi$. With each element $\phi_i \in \Phi$ associate the points $(p_i, q_i)$ that have the same $x$ or $y$ coordinates in $XY(\phi_i)$.

(2) Sort the elements of $\Phi$.

(3) Generate $\chi(\phi_0)$ and $\psi(\phi_0)$ by sorting the points in $S$ in the system $XY$.

(4) **for** $i = 1$ to $k+1$ **do**

    (a) For any $\alpha \in \Delta_i$, generate $\chi(\alpha)$ and $\psi(\alpha)$ from $\chi(\phi_{i-1})$ and $\psi(\phi_{i-1})$, respectively. Use the sweep line technique described in Section 3.1 and lemma 4.2 to find two optimum, parallel, non-overlapping rectangles in the range $[\phi_{i-1}, \phi_i)$.

    (b) Store the optimal pair of rectangles in a list $L$.

    **endfor**

(5) Traverse the list $L$ to find the optimum placement.

### 4.1.1 Analysis

Correctness of the algorithm follows immediately from Lemma 4.1. The set $\Phi$ is generated by all lines joining pairs of points of $S$ and by sorting the elements of $\Phi$, sequence $\Phi$ is generated in $O(n^2 \log n)$ time. The sequences $\chi(\phi_0)$ and $\psi(\phi_0)$ are obtained in $O(n \log n)$ time. At each iteration the sequences $\chi(\alpha)$ and $\psi(\alpha)$ are constructed in $O(1)$ time by constant number of interchanges in the previous sequences. We can locate the optimum pair of parallel non-overlapping rectangles in the system $XY(\phi_i)$ in $O(n)$ time (using sweep line technique). By lemma 4.2, we can also find the optimum placement of a given pair of parallel non-overlapping rectangles, in the interval $\Delta_i$, in $O(1)$ time. It follows that,

**Theorem 4.1** *The problem of finding two optimal, parallel, non-overlapping rectangles enclosing $S$, can be solved in $O(n^3)$ time.*

### 4.2 The rectangles may overlap

In this section, we present an efficient algorithm to locate two parallel, orientation independent, optimal rectangles, with nonempty intersection region.

From the discussion in Section 3.2 it follows that, only the points in $\wp(\alpha)$ decide the location of the optimum pair of parallel rectangles, with nonempty intersection region, in the system $XY(\alpha)$. It is simple to observe that the set $\wp(\alpha)$ is identical for all $\alpha \in \Delta_i$.

Let $B_1$ and $B_2$ be the ordered sets of boundary points (ordered as top, left, bottom and right boundary points) that define the two isothetic overlapping rectangles in the system $XY(\alpha)$ such that, together they cover the set $S$ and none of the boundary points of one rectangle is covered by the other rectangle. The tuple $(B_1, B_2)$ is called a *placement* in the system $XY(\alpha)$. The optimum pair of rectangles define a *placement* in some coordinate system.

**Observation 3** *There are $O(n^2)$ distinct placements in a system $XY(\alpha)$.*

**Observation 4** *The placements in $XY(\alpha)$ are the same for all $\alpha \in \Delta_i$.*

Recall that, $p_i$ and $q_i$ are the points having same $x$ or $y$-coordinates in $XY(\phi_i)$.

**Observation 5** *A placement $(B_1, B_2)$ in a separable interval $\Delta_i$ is also a placement in the interval $\Delta_{i+1}$ (and vice versa), whenever none of the points in $B_1 \cup B_2$ is either $p_i$ or $q_i$.*

Recall (from Section 2) the definition of *extreme points* of $S$ in $XY(\alpha)$ as the ordered set of boundary points $T(\alpha)$ of the smallest axis-aligned rectangle $D(\alpha)$ enclosing $S$. Surely, the extreme points of $S$ remain unaltered for all $\alpha \in \Delta_i$. Let $T_i = T(\alpha)$ be the extreme points of $S$ in $XY(\alpha)$, where $\alpha \in \Delta_i$, $1 \leq i \leq k+1$.

**Lemma 4.3** *Let $p$ be a point in $\wp(\alpha)$, $\alpha \in \Delta_i$, such that $p \notin T_i$. Then, all placements $(B_1, B_2)$ in the interval $\Delta_i$, with $p \in B_1 \cup B_2$, can be identified in $O(n)$ time, provided the sequences $\chi(\alpha)$ and $\psi(\alpha)$ are given.*

*Proof*: As the two rectangles overlap, a pair of sides of each of the two rectangles $D(\alpha, B_1)$ and $D(\alpha, B_2)$ are fixed by the four *extreme points* of $S$ in $XY(\alpha)$. Using the sequences $\chi(\alpha)$ and $\psi(\alpha)$, order the set $\wp(\alpha)$ in such a way that, by fixing one *extreme* point of one of the rectangles at $p$ one can use a sweep line algorithm to position the only other remaining extreme point of the rectangle and construct all the placements, with $p \in B_1 \cup B_2$, in $O(n)$ time. ∎

**Lemma 4.4** *If $T_i = T_{i+1}$, then the number of placements in the interval $\Delta_{i+1}$ that differ from the placements in the interval $\Delta_i$ is at most $O(n)$ and vice versa. Moreover, all these differing placements can be identified in $O(n)$ time.*

*Proof*: Although there are $O(n^2)$ *distinct placements* in any given system $XY(\alpha)$, it is evident that any *placement* $(B_1, B_2)$ in the interval $\Delta_{i+1}$ that

7

differs from all the placements in $\Delta_i$ must have $p_i$ or $q_i$ belonging to $B_1 \cup B_2$ (Observation 5). Assume without any loss in generality that all the four points of $T_i$ are distinct. If none of $p_i$ and $q_i$ belongs to $T_i$ then there are only $O(n)$ *distinct placements* of the form $(B_1, B_2)$ with $p_i$ or $q_i$ belonging to $B_1 \cup B_2$. If $p_i \in T_i$ and $q_i \notin T_i$ then each *placement* $(B_1, B_2)$ in the interval $\Delta_{i+1}$ that differs from all the placements in $\Delta_i$ must have $q_i \in B_1 \cup B_2$. All the *differing placements* in the previous two cases can be identified in $O(n)$ time (lemma 4.3). The only other case is when both $p_i$ and $q_i$ belong to $T_i$. Since $T_i = T_{i+1}$, either $p_i, q_i$ are the topmost and bottommost points of $D(\alpha)$ or the leftmost and rightmost points of $D(\alpha)$. Assume without any loss of generality that $p_i, q_i$ are the topmost and bottommost points of $D(\alpha)$, respectively. Then, the placements $(B_1, B_2)$ that differ between the intervals $\Delta_{i+1}$ and $\Delta_i$ are exactly those where at least one of $p_i, q_i$ is either the left or right extreme point of the rectangle formed by $B_1$ or $B_2$. By an argument similar in spirit to that of lemma 4.3, it can be shown that there are only $O(n)$ such placements all of which can be identified in $O(n)$ time. ∎

**Theorem 4.2** *There is a total of $O(n^3)$ distinct placements in the interval $[0, \frac{\pi}{2}]$, all of which can be identified in $O(n^3)$ time.*

*Proof*: To start with, there are $O(n^2)$ distinct placements in the interval $\Delta_1$. If $T_i = T_{i+1}$, then in the separable interval $\Delta_{i+1}$ an additional of $O(n)$ new placements are introduced (lemma 4.4). The set $T_i$ may differ from $T_{i+1}$ for a maximum of $h$ times, where $h$ is the number of edges of $CH(S)$. Thus, for at most $h$ times $O(n^2)$ new placements are introduced (observation 3). As the total number of separable interval is $O(n^2)$, the total number of distinct placements is bounded by $(O(n^2) - h) \cdot O(n) + h \cdot O(n^2) = O(n^3)$. ∎

An instance can be easily generated where the number of distinct placements in the interval $[0, \frac{\pi}{2}]$ is $\Omega(n^3)$. We are now ready to describe the algorithm for locating an optimum pair of parallel overlapping rectangles.

*4.2.1 The Algorithm*

Simple arrays and lists are sufficient to construct our data structures. An array $M$ stores the set of points in $S$. A point in $M$ (say $p$) that is in $\wp(\alpha)$ for some $\alpha$ in $[0, \frac{\pi}{2}]$, maintains a list structure namely, *placement list* that keeps all placements of the form $(B_1, B_2)$ such that $p \in B_1 \cup B_2$. Each placement $C = (B_1, B_2)$ in a *placement list* associates with it an interval $(\beta_s, \beta_f)$ such that for all $\beta \in (\beta_s, \beta_f)$, $(B_1, B_2)$ is a placement in $XY(\beta)$. We use the variables $C.start$ and $C.finish$ to indicate $\beta_s$ and $\beta_f$, respectively.

**Optimal_overlapping_rectangles**$(S)$

(1) Compute $\Theta$ and generate the sequences $\Phi$, $\chi(\phi_0)$ and $\psi(\phi_0)$ as in Steps 1, 2 and 3 of the algorithm Optimal_disjoint_rectangles.

(2) Initialize the array $M$ and form all the *placement lists* with the *placements* in $XY$. For each placement $C$, initialize $C.start = 0$.

(3) Find the location of the pair of optimum axis-aligned rectangles with nonempty region of intersection in the system $XY$. Store this optimum location along with the area of the larger rectangle in a variable $\mathcal{O}$.

(4) **for** $i = 1$ to $k$ **do**

    (a) Consider coordinate system $XY(\alpha)$ for any $\alpha \in \Delta_{i+1}$. Let $p_i$ and $q_i$ be the points with equal $x$ or $y$-coordinates in the system $XY(\phi_i)$. Generate $\chi(\alpha)$ and $\psi(\alpha)$ from the previous sequences $\chi(\gamma)$ and $\psi(\gamma)$, where $\gamma \in \Delta_i$, and form the set $\wp(\alpha)$.

    (b) **if** $T_i = T_{i+1}$ **then**

        If $p_i, q_i \notin T_i$, delete all the placements from the placement lists of points $p_i, q_i$ and form the placement lists of $p_i$ and $q_i$, anew, using $\wp(\alpha)$, $\alpha \in \Delta_{i+1}$. Else if, $p_i \in T_i, q_i \notin T_i$, delete all the placements from the placement list of point $q_i$ and form the placement list, anew. Else, if $p_i, q_i \in T_i$, deletion and insertion of the placement lists of $p_i$ and $q_i$ are handled in a similar way based on the arguments given in the proof of lemma 4.4.

    **else**

        Delete all the placements from all the placement lists. Form all the placement lists, anew, using the set $\wp(\alpha)$, $\alpha \in \Delta_{i+1}$.

    **endif**

    (c) For each deleted placement $C = (B_1, B_2)$, assign $C.finish = \phi_i$. Solve for $Area(D(\beta, B_1)) = Area(D(\beta, B_2))$ in the range $\beta \in (C.start, C.finish)$, and find the areas of the rectangles $D(\beta, B_1)$ and $D(\beta, B_2)$ at $\beta = C.start$ and $\beta = C.finish$. Update the optimum location in $\mathcal{O}$ by comparing with these locations.

    (d) For each newly formed placement $C$, assign $C.start = \phi_i$.

(5) Delete all the placements from all the placement lists. For each deleted placement $C = (B_1, B_2)$, assign $C.finish = \phi_i$ and solve for $Area(D(\beta, B_1)) = Area(D(\beta, B_2))$ in $\beta \in (C.start, C.finish)$. Also find the locations of the placement $C$ in the systems $XY(C.start)$ and $XY(C.finish)$. Update the optimum location in $\mathcal{O}$, accordingly.

### 4.2.2 Correctness

The algorithm systematically checks all the $O(n^3)$ distinct *placements* defined in the interval $[0, \frac{\pi}{2}]$. The correctness of the algorithm follows from Observation 5, Lemma 4.4 and Theorem 4.2.

### 4.2.3 Complexity

If $T_i = T_{i+1}$, then at each iteration of the for loop at step 4, $O(n)$ new placements are added and $O(n)$ placements are deleted. The processing of each added or deleted placement takes $O(1)$ time. If $h$ is the number of edges of $CH(S)$ then for only $h$ iterations of the loop $O(n^2)$ placements are added

or deleted. Therefore, time complexity of the algorithm is $(\binom{n}{2} - h)O(n) + hO(n^2) = O(n^3)$. Moreover, from Observation 3 we conclude that the space complexity of the algorithm is $O(n^2)$.

**Theorem 4.3** *The problem of locating two parallel rectangles covering a given set of points $S$ in a plane, such that the area of the larger rectangle is minimized, can be solved in $O(n^3)$ time using $O(n^2)$ space.*

## 5   Conclusion

Our algorithm provides a simple way to search all the $O(n^3)$ distinct *placements* in the interval $[0, \frac{\pi}{2}]$ in $O(n^3)$ time. Without altering the complexity, this technique may be used to solve another related problem of covering: finding a cover by two parallel rectangles such that the sum of the areas of the rectangles is minimized. In this case, the only changes in the algorithm are at Step 4(c) and Step 5, where we minimize the function $A(\beta) = Area(D(\beta, B_1)) + Area(D(\beta, B_2)$ by equating the derivative of $A(\beta)$ with respect to $\beta$ to zero in the interval $\beta \in$ *(C.start, C.finish)*. We conclude by suggesting that, it will be interesting to inspect the complexities of these optimization problems when the rectangles *need not* be parallel, i.e. when they are allowed to move freely, independent of one another.

## References

[1]  Sergei Bespamyatnikh and Michael Segal, *Covering a set of points by two axis-parallel boxes*, Information Processing Letters, vol. 75, pp. 95-100, 2000.

[2]  J.W Jaromczyk and M. Kowaluk, *Orientation independent covering of point sets in $R^2$ with pairs of rectangles or optimal squares*, European Workshop on Comp. Geometry, pp. 54-61, 1996.

[3]  M. Katz and K. Kedem and M. Segal, *Discrete rectilinear 2-center problems*, Computational Geometry: Theory and Applications, vol. 15, pp. 203-214, 2000.

[4]  A. Glozman and K. Kedem and G. Shpitalnik, *On some geometric selection and optimization problems via sorted matrices*, Computational Geometry: Theory and Applications, vol. 11, pp. 17-28, 1998.

[5]  H. Freeman and R. Shapira, *Determining the Minimum-Area Enclosing Rectangle for an Arbitrary Closed Curve*, Communications of the ACM, vol. 18, pp. 409-413, 1975.

[6]  M. Hoffmann, *Covering Polygons with Few Rectangles*, 17th European Workshop on Computational Geometry, Berlin, pp. 39-42, 2001.