Separability by two lines and by nearly-straight polygonal chains

Ferran Hurtado[*]   Mercè Mora[*]   Pedro A. Ramos[†]   Carlos Seara[*]

19th February 2003

Abstract

In this paper we study the separability in the plane by two criteria: *double wedge separability* and Θ-*separability*. We give an $O(N \log N)$-time optimal algorithm for computing all the vertices of separating double wedges of two disjoint sets of objects (points, segments, polygons and circles) and an $O((N/\Theta_0) \log N)$-time algorithm for computing a nearly-straight minimal Θ-polygonal chain separating two sets of points, where Θ_0 is a value which depends on the position of the points.

1   Introduction

Let B and R be two disjoint sets of objects in the plane which we denote as *blue* and *red* objects, respectively. The objects we consider are points, segments, polygons and circles. When the objects are polygons we use n and m for the total number of edges of the polygons in B and R, respectively; otherwise n and m are simply the number of objects in B and R. In any case $N = \max\{n, m\}$.

We say that a set $\mathcal{C}$ of curves in the plane separates B from R when every cell of the arrangement induced by $\mathcal{C}$ contains objects only from B or from R. Deciding whether the sets B and R can be separated by means of a single line (a hyperplane in any constant dimension) can be done in $O(N)$ time, which is optimal [12, 13]; this is *linear separability*, which is the simplest kind of geometric separation. The problem of finding the convex polygon with fewest edges separating two point sets in the plane (when this is possible) is solved in [9] in $O(N \log N)$ time. *Circular separability* in the plane can be reduced to linear separability in three dimensions after lifting the points to the unit paraboloid, yet an optimization problem arises, consisting of computing the largest separating circle [4, 5]

Several additional ways of separating two sets have been considered, often trying to use separators conceptually close to the ideal linear separability. In [10] it was proven that finding a polygonal-chain separator with fewest edges (the *minimum-link red-blue separation* problem) is NP-complete. The separation by means of two rays with common apex (*wedge separability*) or by two parallel lines (*strip separability*) is studied in [11]. Both kinds of separability can be decided in optimal $\Theta(N \log N)$ time, and when the answers are affirmative, the wedges with maximum or minimum angle, and the narrowest and widest strip, can be obtained within the same time bound.

---

[*]Dpt. Matemàtica Aplicada II, Universitat Politècnica de Catalunya, Pau Gargallo 5, 08028 Barcelona, Spain, e-mail: {hurtado,mora,seara}@ma2.upc.es. Partially supported by projects DGES-MEC PB98-0933 and Gen. Cat. 2001SGR00224.

[†]Dpto. de Matemáticas, Universidad de Alcalá, Aptdo. de Correos 20, 28871 Alcalá de Henares, Madrid, Spain. pedro.ramos@uah.es. Partially supported by projects DGES-MEC PB98-0933 and PB98-0713-C02-01.

In this paper we study the separability by means of two lines, and by means of a single polygonal chain having a "special simplicity", as defined more precisely next (Figure 1):

Definition 1. *Two disjoint sets of objects B and R are* two lines separable *or* double wedge separable *if there exist two straight lines intersecting at a point p, the vertex of the double wedge, such that there is a non-trivial partition $\{B_1, B_2\}$ of B and a non-trivial partition $\{R_1, R_2\}$ of R such that B_1, R_1, B_2 and R_2 appear angularly in this order in the four wedges produced by the two lines.*

We consider non-trivial partitions for B and R because otherwise double wedge separability is a special case of wedge separability which can be detected directly with the algorithms in [11]. We use the term *double wedge* as in [8], page 362 (Figure 1a). Thus, two intersecting lines define two complementary double wedges. According to this terminology, the sets B and R are double wedge separable if, and only if, there exist two complementary double wedges such that one contains all the red points, the *red double wedge*, and the other contains all the blue points, the *blue double wedge* (Figure 1b). We also say that the red double wedge is the union of two opposite *red wedges*; opposite *blue wedges* are defined analogously from the blue double wedge.

Definition 2. *A Θ-polygonal chain is a polygonal chain such that all of its vertices have angle Θ and it turns alternately left and right. Two disjoint sets of objects B and R are Θ-separable if there exists a Θ-polygonal chain separating B and R.*
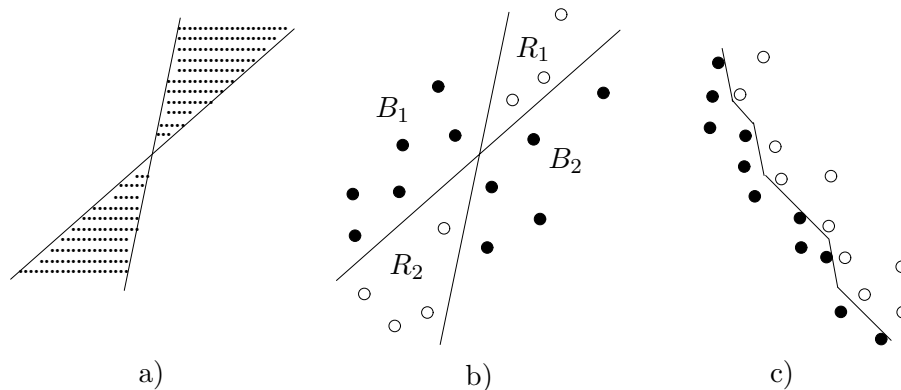

Figure 1: a) Double wedge, b) double wedge separability, c) Θ-separability.

In Section 2 we show that the double wedge separability of two disjoint point sets can be decided in $O(N \log N)$ time, and the algorithm allows the construction of the region formed by the vertices of the separating double wedges. The algorithm is optimal because an $\Omega(N \log N)$ lower bound for the decision problem is shown in [1], and can be extended to objects other than points.

In Section 3 we study the Θ-separability of two disjoint point sets in the plane. We observe that two given sets B and R can always be separated by a Θ-polygonal chain for Θ small enough. However, we are interested in maximizing the angle because the closer to π is the better approximation to linear separability. We have obtained $O((N/\Theta_0) \log N)$ time algorithms for computing separating Θ-polygonal chains with the maximum angle, where Θ_0 is a value which depends on the position of the points and, among these, one with the minimum number of edges.

Many other variations on these problems are systematically studied in [14].

2

2 Double wedge separability

Let B and R be two disjoint point sets in the plane, both of them in general position. This assumption is not essential for our algorithms to work, but handling degeneracies would require the description of many details and would hide the crucial ideas. We also assume hereafter that no double wedge separator, if there is any, has an empty wedge, because this situation is a special case of wedge separability which can be detected directly with the algorithms in [11]. Given a double wedge ω with vertex at p and separating B and R, the plane is decomposed into two complementary double wedges with vertex p, the *red double wedge* and the *blue double wedge*, with aperture angles denoted by α_r and α_b, respectively. We say that a direction is red (blue) if the line through p in that direction is contained in the red (blue) double wedge (the boundary lines get both colors). Because $\alpha_r + \alpha_b = \pi$, either $\alpha_r \leq \pi/2$ or $\alpha_b \leq \pi/2$. Without loss of generality, we assume hereafter that $\alpha_r \leq \pi/2$, hence either the vertical direction is blue or the horizontal direction is blue (or both).

Let Ω be the (possibly empty) set of double wedges separating B and R such that $\alpha_r \leq \pi/2$ and the vertical direction is blue. We show in the rest of this section how to compute the locus of vertices of double wedges in Ω; the total set of all possible vertices of double wedges separating B and R can be computed repeating the process for the rest of the cases.

Observe that any direction defined by two red points in opposite red wedges is a red direction; we can always get one by picking the red points with minimum and maximum abscissa. For that direction, we relabel the points in such a way that $\{r_1, r_2, \ldots, r_m\}$ is monotone in that direction and $\{b_1, \ldots, b_n\}$ is monotone in the perpendicular direction, which is necessarily blue because $\alpha_r \leq \pi/2$. After this relabelling, we immediately get:

Lemma 1. *The wedge partition produced by $\omega \in \Omega$ is given by $B = \{b_1, \ldots, b_i\} \cup \{b_{i+1}, \ldots, b_n\}$ and $R = \{r_1, \ldots, r_j\} \cup \{r_{j+1}, \ldots, r_m\}$ for some $i = 1, \ldots, n-1$, $j = 1, \ldots, m-1$.*

According to the above considerations, for a given input we always start by taking the direction defined by the red points with minimum and maximum abscissa and change the coordinate system for this direction to be horizontal and relabel the red points by increasing abscissa and the blue points by decreasing ordinate. We assume in what follows that this step has already been done.

Computing feasible partitions

In order to compute feasible partitions for B and R into monochromatic opposite wedges, we consider the monotone polygonal chains $\mathcal{P}_B$, with vertices $\{b_1, \ldots, b_n\}$ and $\mathcal{P}_R$, with vertices $\{r_1, \ldots, r_m\}$. Each edge of a polygonal chain induces a partition in the corresponding set given by the vertices of the two chains that appear when the edge is removed.

Lemma 2. *If $\Omega \neq \varnothing$, then there exist two edges $e_r \in \mathcal{P}_R$ and $e_b \in \mathcal{P}_B$ such that $(\mathcal{P}_R \smallsetminus e_r) \cap (\mathcal{P}_B \smallsetminus e_b) = \varnothing$. Furthermore, at least one of the following statements holds:*

(1) There exists exactly one edge $e_r \in \mathcal{P}_R$ having more than one intersection with $\mathcal{P}_B$. In this case, all the double wedges of Ω separate R in the components induced by e_r.

(2) There exists exactly one edge $e_b \in \mathcal{P}_B$ having more than one intersection with $\mathcal{P}_R$. In this case, all the double wedges of Ω separate B in the components induced by e_b.

(3) $\mathcal{P}_R$ and $\mathcal{P}_B$ intersect in one point, $e_r \cap e_b$. In this case, all the double wedges of Ω separate B or R (and maybe both) in the components induced by e_b and e_r, respectively.

Proof. The first claim follows immediately from Lemma 1 observing that, if the partition given by a double wedge ω is $B = \{b_1, \ldots, b_i\} \cup \{b_{i+1}, \ldots, b_n\}$ and $R = \{r_1, \ldots, r_j\} \cup \{r_{j+1}, \ldots, r_m\}$, then the polygonal chains $\{b_1, \ldots, b_i\}$, $\{b_{i+1}, \ldots, b_n\}$, $\{r_1, \ldots, r_j\}$ and $\{r_{j+1}, \ldots, r_m\}$ do not intersect each other. For the second claim, regarding the number and position of the intersections, we observe that there is at most one edge in each polygonal chain having more than one intersection and that, if all the edges have at most one intersection, then there is only one intersection between $\mathcal{P}_R$ and $\mathcal{P}_B$ because the number of intersections between the polygonal chains is odd (Figure 2). Finally, in case (1), as the lines that form the double wedge $\omega \in \Omega$ cannot intersect the chains obtained by removing the edge e_r, ω must separate R precisely in the components induced by e_r; the other two cases are argued similarly. $\square$
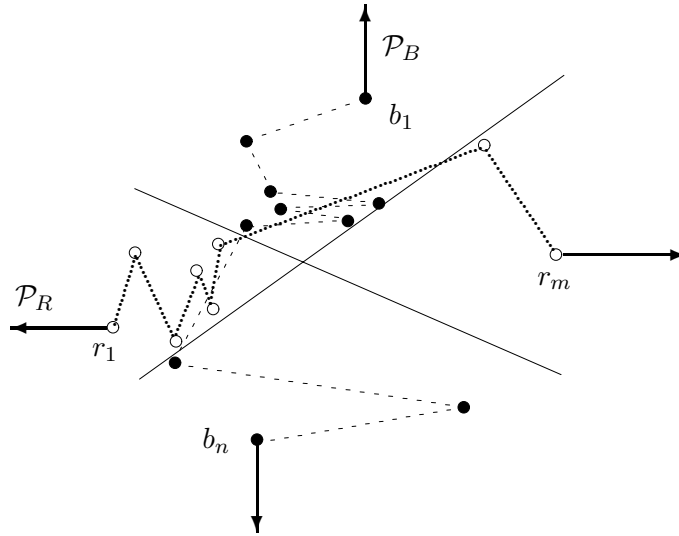

Figure 2: $\mathcal{P}_B$ and $\mathcal{P}_R$.

Let us observe that if $\Omega \neq \varnothing$, then $\mathcal{P}_B$ and $\mathcal{P}_R$ intersect a linear number of times and, therefore, we can use a standard algorithm [3, 6] for computing segment intersections in order to compute candidate partitions of the sets in $O(N \log N)$ time. Moreover, we observe that Lemma 2 implies that there is at most one candidate partition for one of the sets and, therefore, there are $O(N)$ candidate partitions.

Assume that we have a candidate partition $R = \{R_1, R_2\}$ induced by the edge $r_j r_{j+1} \in \mathcal{P}_R$, i.e., $R_1 = \{r_1, \ldots, r_j\}$ and $R_2 = \{r_{j+1}, \ldots, r_m\}$. For $i = 1, \ldots, n-1$, let $B_i = \{b_1, \ldots, b_i\}$ and $B_i' = \{b_{i+1}, \ldots, b_n\}$ be the bipartition of B induced by the edge $b_i b_{i+1}$ of $\mathcal{P}_B$. A necessary condition for the existence of a double wedge separating B and R is that the sets R_1, R_2, B_i and B_i' are pairwise linearly separable. We show next how to compute the tentative partitions fulfilling this condition, assuming that $CH(R_1)$ and $CH(R_2)$ have already been pre-computed.

Lemma 3. *For a fixed partition $R = \{R_1, R_2\}$ of $\mathcal{P}_R$, the list L_B of indices i such that R_1, R_2, B_i and B_i' are pairwise line separable can be computed in $O(n \log m)$ time.*

Proof. We compute first the indices i such that B_i is line separable from R_1 and R_2. Obviously, if B_i is not line separable from R_1 and R_2 then the same is true for B_{i+k}, for $k \geq 1$. If B_i is

line separable from R_1 and R_2, then the linear separability of B_{i+1} can be decided by computing the lines of support of $CH(B_i)$ from b_{i+1}, taking the segments from b_i to the contact points, and checking whether they intersect or not $CH(R_1)$ and $CH(R_2)$. This can be done in $O(\log m)$ time, therefore the list of indices i such that B_i is line separable from R_1 and R_2 can be computed in $O(n \log m)$ time. Within the same time bound, by processing the points from b_n to b_1, the sets B_i' which are line separable from R_1 and R_2 can also be computed. Finally, L_B, i.e., the list of indices i such that both B_i and B_i' are line separable from R_1 and R_2 can be obtained from the two lists in $O(n)$ additional time. □

Lemma 4. *For an index $i \in L_B$ the set of vertices of double wedges separating R and B according to the partition R_1, R_2, B_i and B_i' is a (possibly degenerate) quadrilateral which can be computed in $O(\log N)$ time if the convex hulls $CH(R_1)$, $CH(R_2)$, $CH(B_i)$ and $CH(B_i')$ are given as part of the input as structures allowing binary search.*

Proof. A point p is a vertex of a double wedge separating R_1, R_2, B_i and B_i' if, and only if, p is the intersection point of two lines l_1 and l_2, where l_1 separates $CH(R_1 \cup B_i)$ from $CH(R_2 \cup B_i')$ and l_2 separates $CH(R_1 \cup B_i')$ from $CH(R_2 \cup B_i)$.

The locus of points swept by lines separating two convex polygons is bounded by two concave chains defined by edges of the polygons and by the separating common supporting lines (Figure 3).
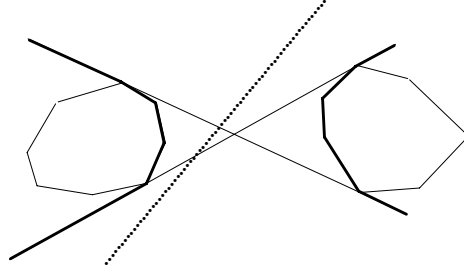

Figure 3: Two concave chains that bound the region of separating lines.

When the two polygons are $CH(R_1 \cup B_i)$ and $CH(R_2 \cup B_i')$, let $\mathcal{C}_{1i}$ be the chain which contains edges from $CH(R_1 \cup B_i)$ and let s_{1i} the only edge in $\mathcal{C}_{1i}$ that is crossed by lines separating R_1 and B_i.

We denote by h_{1i} the half-plane bounded s_{1i} which contains $CH(R_1 \cup B_i)$; h_{2i}, h_{1i}' and h_{2i}' are defined analogously (Figure 4). The set of vertices of double wedges separating R_1, R_2, B_i and B_i' is $h_{1i} \cap h_{2i} \cap h_{1i}' \cap h_{2i}'$. As the four half-planes can be computed in $O(\log N)$ time, the claim is proved. □

The computation in the above lemma has to be performed for every $i \in L_B$. $CH(R_1)$, $CH(R_2)$ are given and $CH(B_i)$ and $CH(B_i')$ can be maintained by mimicking the incremental construction of $CH(B)$ from left to right and reversely, giving an overall running time of $O(n \log n)$.

The previous lemmas combine into the following result:

Theorem 1. *Let B and R be two disjoint sets of points in the plane. The region of vertices of double wedges separating B and R can be computed in $O(N \log N)$ time.*

The above algorithm for computing a description of all the separating double wedges is optimal as an $\Omega(N \log N)$ lower bound for the decision problem of double wedge separability has been shown by Arkin et al. in [1].
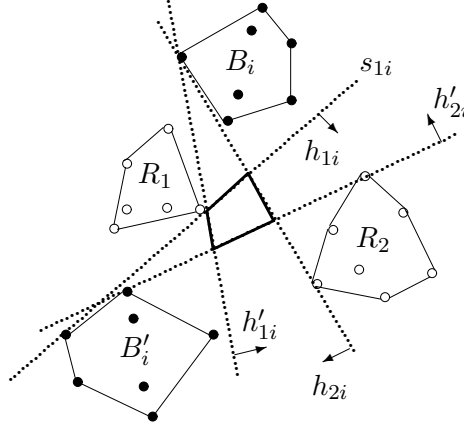
5

Figure 4: Half planes h_{1i}, h_{2i}, h'_{1i} and h'_{2i}.

There are at most $n + m - 2$ combinatorially different ways of separating R and B by means of double wedges, because it can be shown that there are solutions for at most one red partition combined with $n - 1$ different blue partitions and one blue partition combined with $m - 1$ different red partitions. We omit the proof here, it can be found in [14].

2.1 Double wedges with maximum and minimum aperture angle

By *aperture angle* of a double wedge we mean $\max\{\alpha_r, \alpha_b\}$, $\alpha_r + \alpha_b = \pi$. According to this definition, the values of the maximum and the minimum aperture angle are in between $\pi/2$ and π.

- MAXIMUM APERTURE ANGLE: Let l_1 and l_2 be two lines defining a double wedge for the partition R_1, R_2, B_i and B'_i. Let α be the aperture angle of the two lines. Now fixing the line l_1, we rotate the line l_2 adequately increasing the angle α until l_2 becomes a separating common supporting line l'_2 between $CH(R_1 \cup B'_i)$ and $CH(R_2 \cup B_i)$ (Figure 5). Let α_1 be the aperture angle of the two lines l_1 and l'_2. Then, fixing l'_2, we proceed analogously with the line l_1 increasing the aperture angle α_1 until l_1 becomes a separating common supporting line l'_1 between $CH(R_1 \cup B_i)$ and $CH(R_2 \cup B'_i)$ (Figure 5). Therefore it is clear that the maximum aperture angle for a given double wedge partition of B and R corresponds to the maximum aperture angle defined by separating common supporting lines; there are four possibilities which can be computed in $O(\log N)$ time. To compute the total *maximum aperture angle* we maintain the maximum aperture angle for the double wedge partitions of B and R.

- MINIMUM APERTURE ANGLE: We distinguish between two cases: either the minimum aperture angle is strictly greater than $\pi/2$ or the minimum aperture angle is exactly $\pi/2$. In the first case, we can proceed analogously as in the case of maximum aperture angle but decreasing the aperture angle defined by the two lines l_1 and l_2. Thus, the minimum aperture angle for a given double wedge partition of B and R corresponds to the minimum aperture angle of separating common supporting lines; there are four possibilities which can be computed in $O(\log N)$ time. To compute the total *minimum aperture angle* we maintain the minimum aperture angle for the double wedge partitions of B and R.

  In the second case, assume that the minimum aperture angle $\pi/2$ is given by the perpendicular lines l_1 and l_2. We can translate the two lines in parallel to themselves until they bump into
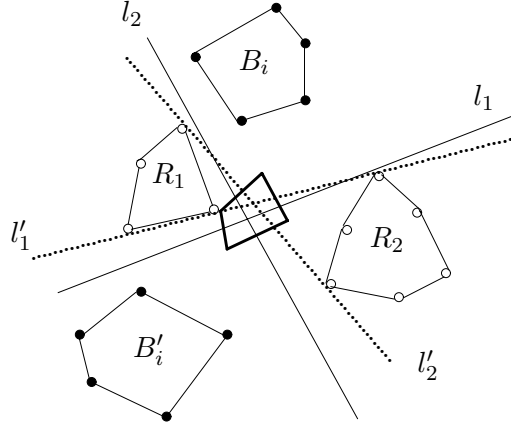
Figure 5: Maximum aperture angle.

points in $CH(R_1)$, $CH(R_2)$, $CH(B_i)$ or $CH(B'_i)$; then we rotate the two lines simultaneously over those convex hulls, maintaining the perpendicularity, until at least one of the two lines (or both) becomes a separating common supporting line between $CH(R_1 \cup B_i)$ and $CH(R_2 \cup B'_i)$ or a separating common supporting line between $CH(R_1 \cup B'_i)$ and $CH(R_2 \cup B_i)$. Now, in order to check whether the minimum aperture angle for a given partition of B and R is exactly $\pi/2$, we compute the four possible common supporting lines and for each one, l, we check in $O(\log N)$ time whether there exists a line perpendicular to l forming a separating double wedge. Once this is true for some double wedge partition of B and R, we can conclude that the minimum aperture angle is exactly $\pi/2$. Therefore we obtain:

Proposition 1. *If the sets B and R are double wedge separable, then the double wedges with maximum and minimum aperture angle can be computed in $O(N \log N)$ time.*

2.2 Separating segments by double wedges

Let S_B and S_R two disjoint sets of n and m segments in the plane classified as red and blue segments respectively. As for points, we consider whether there exists a proper double wedge separating S_B and S_R: each wedge contains only monochromatic segments and each wedge contains at least one segment. The problem is not equivalent to the separability of the endpoints, because a blue wedge, say, might be crossed by a red segment with endpoints on the opposite red wedges. We again look for a separating double wedge where the vertical direction is blue; other cases are handled similarly.

As a first step we consider the double wedge separation of the blue and red sets of endpoints. Suppose that the red endpoints are sorted by x-coordinate and that the blue endpoints are sorted by y-coordinate. We construct the red and blue polygonal chains joining endpoints as we did for sets of points. Only edges of the red polygonal chain, bridging red segments in such a way that their vertical projection does not overlap projections of any red segments, have to be considered as candidates for defining a red partition. We call these edges *critical red bridges* (Figure 6). *Critical blue bridges* are similarly defined. The algorithm for point sets can now be used for solving the problem, but the computation is only required for partitions B_i and B'_i corresponding to critical blue bridges.

Theorem 2. *Let S_B and S_R be disjoint sets of segments in the plane. Whether they are double*
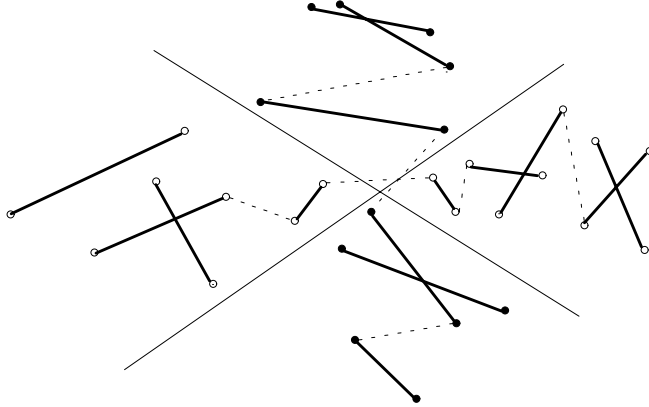
7

Figure 6: Double wedge for segment sets, the dashed segments are the critical red/blue bridges.

wedge separable can be decided in $O(N \log N)$ time. The locus of vertices of separating double wedges and the separating double wedges with maximum and minimum aperture angle can be computed within the same time bound.

Let us remark that this result also settles the separability by double wedges of sets of red and blue polygons.

2.3 Separating circles by double wedges

We consider now the same problem for two disjoint sets C_B and C_R of n and m blue and red circles of the plane respectively (Figure 7). We show next how to adapt to this case the algorithm for segments.

Let $b_1, \ldots, b_n$ and $r_1, \ldots, r_m$ be the centers of the blue and red circles, after relabelling as we did for points (in particular the x axis is the line passing through r_1 and r_m). Replace each circle by its vertical and horizontal diameters in order to obtain as above critical red or blue bridges. Once these critical edges are available, the circles are recovered, as hulls of their union have to be incrementally maintained (as for points), which can de done in $O(N \log N)$ time using the algorithm in [7]. Therefore we obtain:
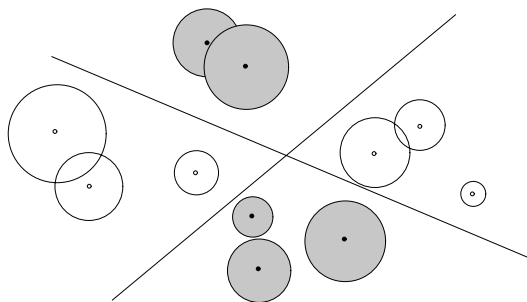

Figure 7: Double wedge for sets of circles.

Theorem 3. *Let C_B and C_R be disjoint sets of circles in the plane. Whether they are double wedge separable can be decided in $O(N \log N)$ time. The locus of vertices of separating double wedges and the separating double wedges with maximum and minimum aperture angle can be computed within the same time bound.*

3 Θ-separability

When the sets B and R are not linearly separable we may be interested in finding a polygonal chain separating the sets and satisfying some restrictions. The problem of computing a polygonal separator with the minimum number of edges is NP-complete [10]. A simpler problem is the Θ-*separability* problem, i.e., determining the separability by a polygonal chain that turns alternately left and right with angle Θ, a Θ-*polygonal chain*. In particular, we are interesting in maximizing the angle Θ of a separating Θ-polygonal chain because the closer to π is the better approximation to linear separability; in fact, linear separability corresponds to π-separability .

It is easy to see that two point sets B and R are always separable by some Θ-polygonal chain with angle Θ small enough. In fact, our first objective is computing an angle Θ_0 such that we can easily guarantee the existence of a Θ_0-polygonal chain separating B and R because this is a necessary step for our later addressing the problem of computing a Θ-polygonal chain separator with maximum angle.

Computing a Θ_0-polygonal chain separator

Let us consider Θ-polygonal chains such that the bisector of the angle Θ is a vertical half-line pointing up and where the red points are above the polygonal chain.

We sort the (red and blue) points by x-coordinate. Without loss of generality we can assume that there are no points with the same x-coordinate; otherwise, this can be easily achieved in $O(N \log N)$ time by rotating the coordinate system.

For every red point r_i let s_i the upwards vertical ray with origin at r_i. We compute an angular value as follows:

1. Let U_b be the upper convex hull of the blue points, having b_1 and b_n as leftmost and rightmost points, respectively.

2. If a red point r_i is below U_b, let us consider the intersection points u and v of the vertical lines through the previous and next blue points with U_b. Let θ_{i1} (θ_{i2}) be the angle defined by the rays s_i and $\overrightarrow{r_i u}$ ($\overrightarrow{r_i v}$) (Figure 8).

3. If r_i is not below U_b we rotate counterclockwise s_i until either we hit a blue point or the angular value $\pi/2$ is reached; we set θ_{i1} to be this angle of rotation. The value of θ_{i2} is similarly defined.

4. Let $\Theta_0 = 2 \min\{\theta_{11}, \theta_{12}, \ldots, \theta_{m1}, \theta_{m2}\}$.

Clearly the vertical wedge with apex at r_i, symmetry axis s_i and aperture angle Θ_0 is empty of blue points, for every r_i. Therefore the lower envelope of all this wedges is a Θ_0-polygonal chain separating B and R. This envelope can be computed by divide and conquer, hence we obtain the following result:

Proposition 2. *Any two disjoint point sets B and R in the plane are Θ_0-separable by some Θ_0-polygonal chain, and one such separator can be constructed in $O(N \log N)$ time.*
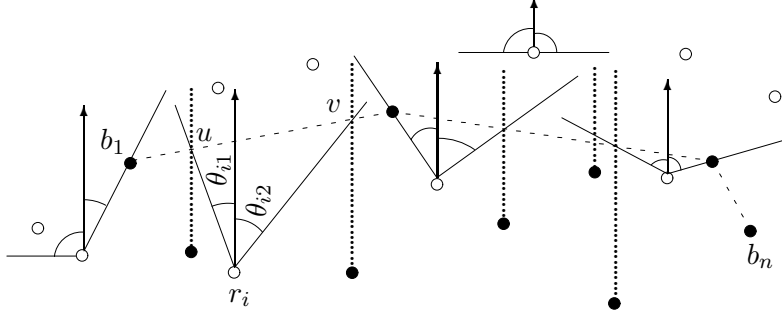
Figure 8: Computing Θ_0.

3.1 Computing a separating Θ-polygonal chain with maximum Θ

A fundamental tool for solving this problem is Theorem 4 by Avis et al. about the computation of the so-called unoriented Θ-maxima points of a planar point set S.

Definition 3. *A ray from a point $p \in S$ is called a* maximal ray *if it passes through another point $q \in S$. A cone is defined by a point p and two rays C and D emanating from p (Figure 9).*

Definition 4. *A point $p \in S$ is an unoriented Θ-maximum with respect to S if, and only if, there exist two maximal rays, C and D, emanating from p with an angle at least Θ between them so that the points of S lie outside the (Θ-angle) cone defined by p, C and D (Figure 9).*
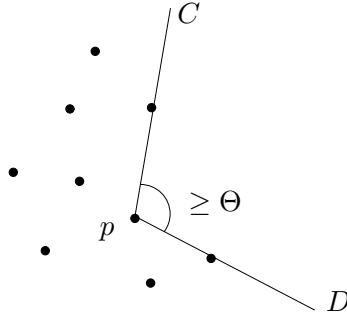

Figure 9: Unoriented Θ-maximum.

Theorem 4. [2] *Let S be a set of n points in the plane. All unoriented Θ-maxima points of S for $\Theta \geq \pi/2$ can be computed in $O(n \log n)$ time and $O(n)$ space. For angles $\Theta < \pi/2$ the Θ-maxima points of S can be computed in $O((n/\Theta) \log n)$ time. The algorithm is optimal for fixed values of Θ.*

The $\pi/2$ constant of the theorem can be substituted by an arbitrary $\Theta_0 > 0$ without changing the asymptotic complexity of the algorithm as a function of n.

The reason for the above results to be crucial for us comes from the observation that if there is a Θ-polygonal chain separating B from R, then all the points in R are "maximal" in the sense

10

that all them are apices of wedges of aperture Θ, with sides parallel to the edges in the separator, and free of points from B (Figure 10).
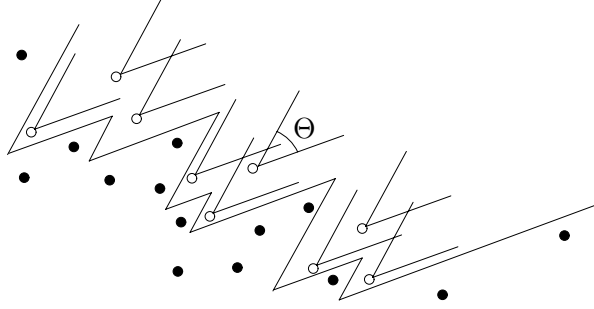

Figure 10: All the red points are unoriented Θ-maximum with respect to B.

First of all we adapt to the bichromatic situation the definition of *unoriented Θ-maximum* of a set of points.

Definition 5. *Let B and R be disjoint sets of blue and red points in the plane. A point $r \in R$ is an* unoriented Θ-maximum with respect to B *if, and only if, there exist two maximal rays, C and D, emanating from r and with an angle at least Θ between them so that no points from B lie inside the (Θ-angle) cone defined by r, C and D (Figure 11).*
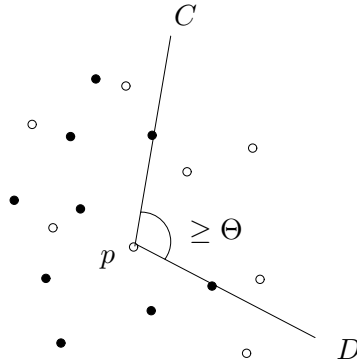

Figure 11: Unoriented Θ-maximum with respect to B.

With this definition it is clear that once we have computed a Θ_0-polygonal chain as in Proposition 2 then all the red points are unoriented Θ_0-maxima with respect to B. Now, we want to compute the maximum angle $\Theta \geq \Theta_0$ such that all the red points are unoriented Θ-maxima with respect to B. As we are making heavy use of the result and algorithm mentioned in Theorem 4, for the sake of clarity we sketch next the two basic steps of that algorithm:

- Procedure CANDIDATES: the input is a planar point set S and an angle β; the output gives the list of edges of the convex hull of S, $CH(S)$, together with a list of the candidate points for each edge. A point p is a candidate for the edge $e = xy \in CH(S)$ if the angle between

the rays $\overrightarrow{px}$ and $\overrightarrow{py}$ is not smaller than β. Notice that if β is less than $\pi/2$, the output of this procedure can have size $\Omega(n/\beta)$.

- Procedure UNORIENTED-MAXIMA: the input is the list of candidates for every edge e of $CH(S)$; the output is the list of unoriented β-maxima points that are apices of wedges that have bounding rays crossing e and aperture angle at least β. For every such maximal point p the output also contains the two rays L_p and R_p defining the widest empty wedge from p. Let $R_{p,e}$ be the ray which is perpendicular to e and has origin at p. For $\beta \geq \frac{\pi}{2}$, ray $R_{p,e}$ must be between L_p and R_p; the procedure sorts the candidate points by their orthogonal projection onto e, and then does two sweeps by lines perpendicular to e, utilizing an incremental convex hull computation, yielding the rays L_p and R_p as a by-product (Figure 12). In case that $\beta < \frac{\pi}{2}$, the angle between L_p and R_p must contain $R_{p,e}$ or one of the $\frac{\pi}{\beta} - 2$ directions which are separated from $R_{p,e}$ by integer multiples of β and the above procedure is executed $(\frac{\pi}{\beta} - 1)$ times, one for each such direction, giving an overall $O((N/\beta)\log N)$ running time.
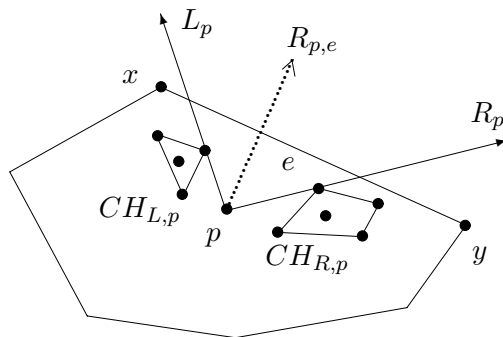

Figure 12: Computing unoriented Θ_0 maxima from candidates.

Remember that all the red points are unoriented Θ_0-maxima with respect to B, for the value Θ_0 previously obtained according to Proposition 2. The algorithm for computing a separating Θ-polygonal chain with maximum Θ has two main parts. First, the algorithm from [2] we have just described is adapted to obtain for every red point r *all* the maximal wedges with apex at r that make r Θ_0-maximum with respect to B. The output is used in the second part for the maximization of the angle Θ in any separating Θ-polygonal chain.

Procedure Θ-POLYGONAL-CHAIN
Input: B, R, Θ_0,
Output: A separating Θ-polygonal chain with maximum $\Theta = \Theta_M$.

1. Compute $CH(B)$, let $\{e_1, \ldots, e_l\}$ be the sorted list of edges of $CH(B)$. Classify the red points into R_I and R_E, where R_I are those that are *interior* to $CH(B)$ and R_E are those that are *exterior or on the boundary* of $CH(B)$.

2. For points in R_I: run the algorithm of Theorem 4 [2] on $B \cup R_I$ to find the maximum angle $\Theta \geq \Theta_0$ such that all the points from R_I are unoriented Θ-maxima with respect to B.

   Procedure CANDIDATES on $B \cup R_I$ gives the list of points from $B \cup R_I$ that are candidates for each edge of $CH(B)$. A point can be a candidate for a constant number of edges: if $\Theta_0 \geq \frac{\pi}{2}$

for at most three edges, if $\Theta_0 < \frac{\pi}{2}$ for at most $\frac{2\pi}{\Theta_0}$. Notice that if Θ_0 is less than $\pi/2$, the output of this procedure can have size $\Omega(n/\Theta_0)$.

Procedure UNORIENTED-MAXIMA above is modified in such a way that the double sweep computes incrementally the blue convex hull which is maintained in a structure allowing logarithmic dynamic maintenance when a blue point is encountered and support line computation when a red point is found. The output is the list of red points that are unoriented β-maxima with respect to B for some $\beta \geq \Theta_0$ and all wedges that have them as apices, aperture angle at least Θ_0 and are free of blue points. The procedure is executed $(\frac{\pi}{\Theta_0} - 1)$ times for each edge, which gives an overall $O((N/\Theta_0)\log N)$ running time.

3. For points $r_i \in R_E \cup B$, we proceed as in step 2 and in addition compute the rays from r_i that support $CH(B)$, because the external angles β_i they define (always greater or equal than π) are also making these points unoriented maxima (Figure 13).
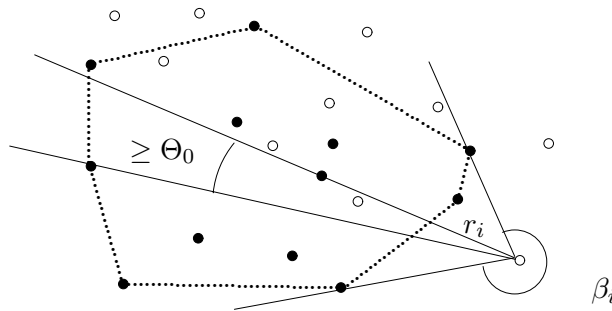

Figure 13: External angle β_i for a point $r_i \in R_E$.

4. For each red point we have a constant number of at most $\frac{2\pi}{\Theta_0} + 1$ angular windows making the point maximal with respect to B and aperture angle at least Θ_0. We place these values as arcs on a circle which has been drawing using several concentric circles, for clarity (Figure 14). The intersection of the arcs (or angular intervals) can be easily obtained in $O(N \log N)$ time and we get a constant number of angular intervals with apertures $\theta_i \geq \Theta_0$ making all red points maximal. Let Θ_M be the maximum of the angles. Notice that Θ_M might appear several times, at most a constant number k. Let $\{I_1, \ldots, I_k\}$ be these Θ_M-size intervals. We describe each I_i by the direction d_i of its bisector, obtaining a constant number of directions $\{d_1, \ldots, d_k\}$.

5. We construct a Θ_M-polygonal chain for d_1 in $O(N \log N)$ time analogously as we did in Proposition 2. (Polygonal chains for all $d_1, d_2, \ldots, d_k$ may be constructed within the same time.)

The correctness of the procedure follows from the preliminary lemmas; the $O((N/\Theta_0)\log N)$ running time dominates the steps and we can state the following theorem:

Theorem 5. *Given two disjoint point sets B and R in the plane, the computation of a separating Θ-polygonal chain with maximum $\Theta = \Theta_M$ can be achieved in $O((N/\Theta_0)\log N)$ time.*

Remark. Notice that the complexity of the preceding algorithm depends on the parameter Θ_0, which unfortunately might be very small as it depends on the particular point sets. However, if
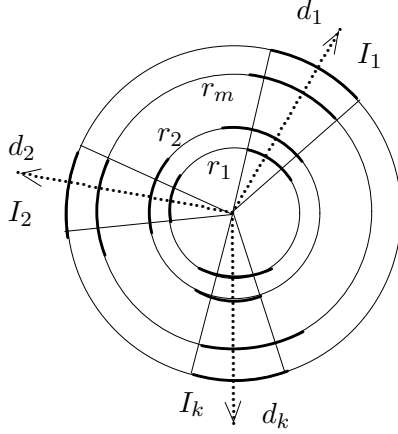
Figure 14: Angular intervals and directions of Θ-polygonal chains.

one is interested only in checking whether the Θ-separability of the two point sets is "close" to π-separability in a precise sense, the preceding approach can be slightly modified while achieving an $O(N \log N)$ running time.

Let us assume, for example, that we are interested in checking whether $2\pi/3$-separability is achievable or not, and only in the affirmative we would look for Θ_M. Now, if the answer is positive, at least one of the four values of the parameter Θ_0 obtained by executing the algorithm in Proposition 2, in the four directions of the coordinate axis (positive and negative) must be greater than $\frac{2\pi}{3} - \frac{\pi}{2}$. This condition can be checked in time $O(N \log N)$, and only when fulfilled we would run the algorithm leading to theorem 5, with complexity $O(N \log N)$.

3.2 Maximum angle and minimum number of vertices

As we have seen in the preceding subsection we might have more than one separating Θ-polygonal chain with maximum $\Theta = \Theta_M$. A natural problem is to require additionally the number of vertices to be minimized. This is the problem we address now; the solution will be called the *max-angle min-vertex* polygonal chain.

From the previous algorithm we have a constant number of directions $\{d_1, \ldots, d_k\}$ corresponding to the bisectors of different solutions (in the sense that they correspond to different angular windows). We show next how to solve the problem for one of them, the procedure is then repeated for the other while keeping the best solution found.

Assume without loss of generality that d_1 is the vertical direction. Let $\mathcal{P}_{\Theta_M, R}$ be the separating Θ_M-polygonal chain with "valley" vertices on red points (Figure 15) obtained by applying the construction from Proposition 2. Let $\mathcal{P}_{\Theta_M, B}$ be the separating Θ_M-polygonal chain with "top" vertices on blue points (Figure 15) obtained by applying the construction from Proposition 2.

Notice that $\mathcal{P}_{\Theta_M, R}$ and $\mathcal{P}_{\Theta_M, B}$ do not cross but they have at least two edges overlapping, due to the maximality of Θ_M.

Lemma 5. *There exists a separating Θ_M-polygonal chain with minimum number of vertices (among those having bisector d_1) such that its edges lie alternately on $\mathcal{P}_{\Theta_M, R}$ and $\mathcal{P}_{\Theta_M, B}$.*

Proof. Let $\mathcal{P}_M$ be a separating Θ_M-polygonal chain with minimum number of vertices for the direction d_1, which will consist of two half-lines and a set of edges. Assume without loss of generality
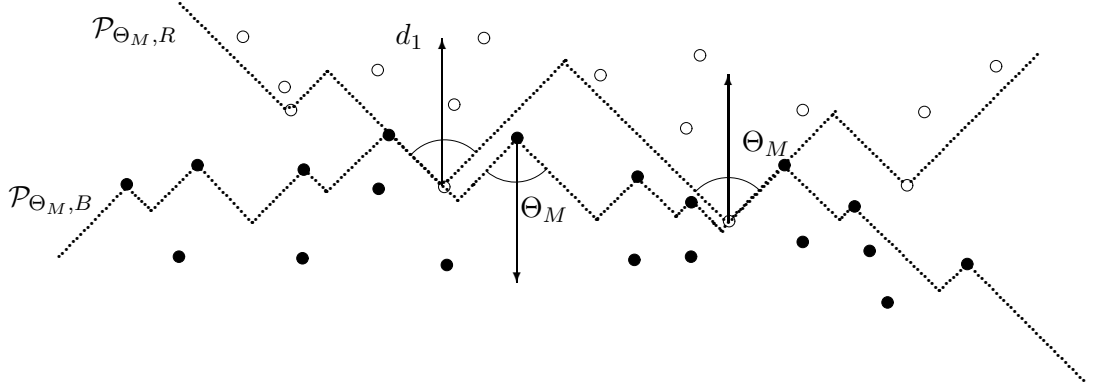
14

Figure 15: Minimizing the number of vertices.

that the leftmost half-line e from $\mathcal{P}_M$ is parallel to the leftmost half-line in $\mathcal{P}_{\Theta_M,R}$: we "push" e, with all its intermediate steps being parallel to the original position, until it touches $\mathcal{P}_{\Theta_M,R}$, then the next edge is similarly pushed to $\mathcal{P}_{\Theta_M,B}$ and so on. Notice that no edge of the current polygonal chain can disappear during this process because of the assumption of initial minimality. In the end, the resulting polygonal chain fulfills the conditions stated in the lemma. □

The above Lemma provides the following greedy approach to find a separating Θ_M-polygonal chain with minimum number of vertices for the direction d_1:

Greedy algorithm

1. Let e the left half-line of $\mathcal{P}_{\Theta_M,R}$.

2. Extend e in the sense of increasing abscissa until it hits $\mathcal{P}_{\Theta_M,B}$, then make a turn of angle Θ_M and extend an edge until hitting $\mathcal{P}_{\Theta_M,R}$ and keep going until the extension of a growing edge goes to infinity.

3. Repeat steps 1 and 2 for the other three extreme half-lines, and exit with the solution giving the smallest number of vertices among these four possibilities.

We know that $\mathcal{P}_{\Theta_M,B}$ and $\mathcal{P}_{\Theta_M,R}$ can be constructed in $O((N/\Theta_0)\log N)$ time, and it is easy to see that the final greedy algorithm requires only additional linear time. As this process is executed a constant number of times (once per each bisector d_i), we can state the following theorem:

Theorem 6. *Given two disjoint point sets B and R in the plane the max-angle min-vertex polygonal chain separating the sets can be computed in $O((N/\Theta_0)\log N)$ time.*

Remark. Notice that again the complexity of the preceding algorithm depends on the parameter Θ_0. However, if one is interested only in computing the max-angle min-vertex polygonal chain in the sense of the remark above then, the time complexity of theorem 6 is $O(N\log N)$.

3.3 Lower bound

Avis et al. established an $\Omega(n\log n)$ lower bound for the computation of the unoriented Θ-maxima of a planar point set S, for $\pi/2 \leq \Theta \leq \pi$ [2]. We adapt next their construction to the computation of the separating Θ-polygonal chain with maximum angle.

Theorem 7. *The problem of computing the Θ-polygonal chain with maximum $\Theta = \Theta_M$ separating two disjoint point sets in the plane, with $\pi/2 \leq \Theta_M < \pi$, has complexity $\Omega(N \log N)$ under the algebraic computation tree model.*

Proof. We use a reduction from integer element uniqueness as in [2]; this problem has a lower bound $\Omega(N \log N)$ under the algebraic computation tree model as proved by Yao [15].

Let $M = \{x_1, \ldots, x_N\}$ be a set of integers. For each x_i, we construct a red point $(i + \epsilon, (Nx_i)^2)$ and five blue points $(i + \epsilon, (Nx_i)^2 + \epsilon)$, $(i + \epsilon, (Nx_i)^2 - \epsilon)$, $(i - \epsilon, (Nx_i)^2)$, $(i - \epsilon, (Nx_i)^2 + \epsilon)$, $(i - \epsilon, (Nx_i)^2 - \epsilon)$, where $\epsilon = 1/4$. Let R and B be the sets of red and blue points obtained by union of these sets for $i = 1, \ldots, N$.
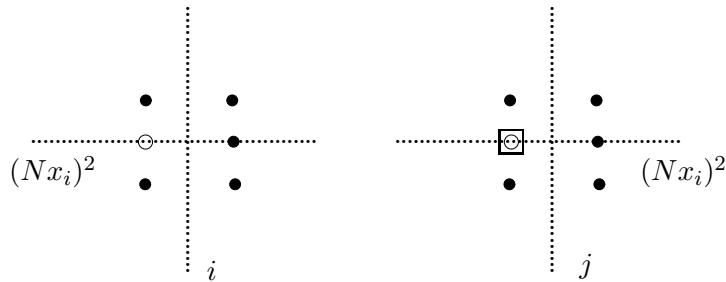

Figure 16: Situation for the case $x_i = x_j$.

If $x_i = x_j$ then at least one red point out of the two red associated points is not a Θ-unoriented maxima with respect to B (Figure 16). Hence, the sets R and B are not separable by any Θ-polygonal chain with $\pi/2 \leq \Theta < \pi$ because in that situation all red points would be Θ-unoriented maxima with respect to B.

Conversely, if x_i is unique in M, then the six points associated to x_i are Θ-unoriented maxima when the colors are disregarded, and the associated red point is Θ-unoriented maximum with respect to B with an angle Θ, $\pi/2 \leq \Theta < \pi$ having as bisector the direction $\vec{v} = (-1, 0)$. Therefore if all the x_i are different then there is a separating Θ-polygonal chain with $\pi/2 \leq \Theta < \pi$ and having as bisector the direction $\vec{v} = (-1, 0)$.

Hence all the elements x_i's in M are distinct if, and only if, there exists some separating Θ-polygonal chain with $\pi/2 \leq \Theta < \pi$, and the claimed reduction is proved. $\square$

References

[1] E. M. Arkin, F. Hurtado, J. S. B. Mitchell, C. Seara, S. S. Skiena, *Some Lower Bounds on Geometric Separability Problems*, 11th Fall Workshop on Computational Geometry, 2001.

[2] D. Avis, B. Beresford-Smith, L. Devroye, H. Elgindy, E. Guévremont, F. Hurtado, B. Zhu, *Unoriented Θ-Maxima in the Plane: Complexity and Algorithms*, SIAM Journal on Computing, Vol. 28, No. 1, 1999, pp. 278–296.

[3] J. L. Bentley, T. A. Ottmann, *Algorithms for Reporting and Counting Geometric Intersections*, IEEE Transactions on Computers, Vol. 28, 1979, pp. 643–647.

[4] B. K. Bhathacharya, *Circular Separability of Planar Point Sets*, Computational Morphology, G. T. Toussaint Ed. North Holland, Amsterdam, 1988.

[5] J.-D. Boissonnat, J. Czyzowicz, O. Devillers, J. Urrutia, M. Yvinec, *Computing Largest Circles Separating Two Sets of Segments*, International Journal of Computational Geometry & Applications, Vol. 10, No. 1, 2000, pp. 41–53.

[6] B. M. Chazelle, H. Edelsbrunner, *An Optimal Algorithm for Intersecting Line Segments*, Proceedings of 29th Annual IEEE Symposium on Foundations of Computer Science, 1988.

[7] O. Devillers, M. J. Golin, *Incremental Algorithms for Finding the Convex Hulls of Circles and the Lower Envelopes of Parabolas*, Proceedings of the Sixth Canadian Conference on Computational Geometry, 1994, pp. 153–158.

[8] H. Edelsbrunner, *Algorithms in Combinatorial Geometry*, vol. 10 of EATCS Monographs in Theoretical Computer Science, Springer-Verlag, 1987.

[9] H. Edelsbrunner, F. P. Preparata, *Minimum Polygonal Separation*, Information and Computation, 77, 1988, pp. 218–232.

[10] S. Fekete, *On the Complexity of Min-Link Red-Blue Separation*, Manuscript, 1992.

[11] F. Hurtado, M. Noy, P. A. Ramos, C. Seara, *Separating Objects in the Plane with Wedges and Strips*, Discrete Applied Mathematics, Vol. 109, 2001, pp. 109–138.

[12] M. E. Houle, *Algorithms for Weak and Wide Separation of Sets*, Discrete Applied Mathematics, Vol. 45, 1993, pp. 139–159.

[13] N. Megiddo, *Linear-time Algorithms for Linear Programming in $\mathbb{R}^3$ and Related Problems*, SIAM Journal on Computing, Vol. 12, No. 4, 1983, pp. 759–776.

[14] C. Seara, *On Geometric Separability*, Ph. D. Thesis, Univ. Politècnica de Catalunya, June 2002.

[15] A. C. C. Yao, *Lower Bounds for Algebraic Computation Trees with Integer Inputs*, Proceedings of 30th Annual IEEE Symposium on Foundations of Computer Science, 1989, pp. 308–313.