Separating objects in the plane by wedges and strips *

(Extended abstract)

Ferran Hurtado† Marc Noy† Pedro A. Ramos‡ Carlos Seara†

Abstract

In this paper we study the separability of two disjoint object sets in the plane by two criteria: we want to know if there is a wedge or a strip separating the object sets. In the affirmative case, we give algorithms for computing all the separating wedges and strips, the wedges with the maximum and minimum angle and the narrowest and the widest strip. The objects we consider are points, segments, polygons and circles.

1 Introduction

Let P and Q be two disjoint object sets in the plane classified as *blue* and *red* objects respectively. The objects we consider are points, segments, polygons and circles. If the objects are polygons, n and m represent the total number of segments of the polygons in P and in Q. In other cases n and m are the number of objects of P and Q respectively, and in any case $N = \max\{n, m\}$.

Let $\mathcal{C}$ be a family of curves in the plane. The sets P and Q are $\mathcal{C}$ *separable* if there exists a separator $S \in \mathcal{C}$ such that every connected component of $\mathbb{R}^2 - S$ contains objects only from P or from Q. If S is a straight line, the sets P and Q are *line separable*. P and Q are line separable if, and only if, their convex hulls do not intersect [16]. The decision problem of linear separability for any of the above object classes can be solved in $O(N)$ time [7,10]. The region of the plane formed by the points of the separating lines can be computed in $O(N \log N)$ time.

There are other separability criteria. For example, in [4] the authors consider the problem of finding the minimum (in the number of edges) convex polygon separating two point sets in the plane. In image processing efficient algorithms to find circular separators for two sets of points can be used to recognize disks [1,5,11]. In this paper we present two criteria of separability: the *wedge separability* and the *strip separability*.

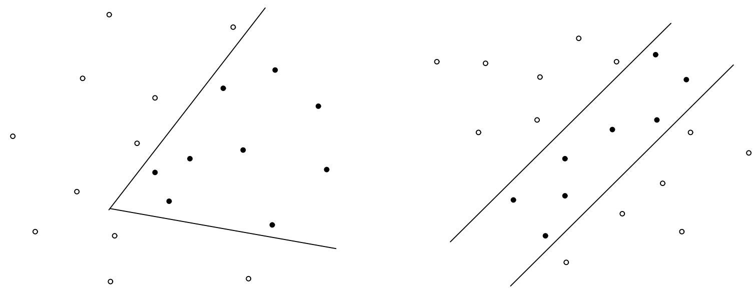
Definition 1. *Two disjoint object sets P and Q are wedge separable if there exists a wedge that contains only all the objects of one of the sets (Figure 1 a)).*

The vertex of the wedge is the common extreme of the half lines. If the angle of the wedge is exactly π, we have the linear separability. Given P and Q, we study the problem of deciding if the sets can be wedge separable and, in the affirmative case, we compute the region of the plane formed by the vertices of separating wedges and additionally, the wedges with the maximum and minimum angle.

Definition 2. *Two disjoint object sets P and Q are strip separable if there exist two parallel straight lines (a strip) that contain only all the objects of one of the sets in between (Figure 1 b)).*

If P and Q are strip separable then they are also wedge separable: just move a little bit one of the parallel lines. Deciding if P and Q are strip separable implies finding a *strip slope*, i.e., the slope of the parallel lines of some strip. Given P and Q, we study the problem of deciding if they can be strip separable and, in the affirmative case, we compute the set of intervals of strip slopes and additionally, the narrowest and the widest strip.

*Partially supported by Proyecto UPC PR-9410. Partially supported by Proyecto DGES-MEC PB96-0005-C02-02.
†Dpt. de Matemàtica Aplicada II, Universitat Politècnica de Catalunya, e-mail:{hurtado, noy, seara}@ma2.upc.es
‡Dpto. de Matemática Aplicada, Facultad de Informática, Universitad Politécnica de Madrid, e-mail:{pramos}@fi.upm.es



a) b)

Figure 1: a) Wedge separability, b) strip separability.

In this paper we show algorithms that solve the wedge and strip separability problems in $O(N \log N)$ time. In Section 2 we study the wedge separability problem for different kind of objects and in Section 3 we study analogously the strip separability problem. The objects are classified as *red* and *blue* objects, a common wise terminology used in many problems.

2 Wedge separability

2.1 Separating points by wedges

Let $P = \{p_1, p_2, \ldots, p_n\}$ and $Q = \{q_1, q_2, \ldots, q_m\}$ be two disjoint point sets in the plane. If $n = 1$ or $m = 1$, P and Q are wedge separable. If $n = 2$ or $m = 2$, for example $Q = \{q_1, q_2\}$, there is a separating wedge that contains Q if, and only if, the segment q_1q_2 does not contain points of P (see Figure 2).

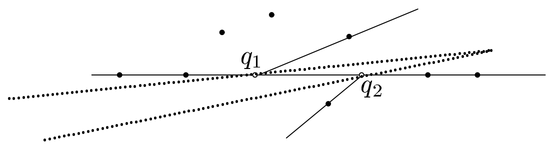


Figure 2: $m = 2$.

Suppose that $n, m \geq 3$ and both P and Q have at least three non collinear points, otherwise the problem can be reduced to the case $n = 2$ or $m = 2$. If the convex hulls of P and Q, $CH(P)$ and $CH(Q)$, do not intersect, the point sets are line separable and therefore wedge separable. A necessary condition for the existence of wedge separability is that the convex hull of one set has to be monochromatic, i.e., it contains only points of one color. In $O(N \log N)$ time we compute the convex hull of each set and determine if any of them is monochromatic, otherwise they are not wedge separable. Suppose that the sets are not line separable and $CH(Q)$ is monochromatic. The problem is to compute the region of the plane formed by vertices of separating wedges.

Tracing the supporting lines from $p_i \in P$ to $CH(Q)$ we obtain two regions A_i and B_i (see Figure 3) that do not contain vertices of separating wedges, otherwise the wedges would contain the point p_i. It is clear that we have to compute $\overline{A} \cap \overline{B}$, where $A = \cup_{i=1}^{n} A_i$ and $B = \cup_{i=1}^{n} B_i$. Next, we describe how to do it.

1) Computing $\overline{A}$

The region $\overline{A}$, as we will see, is a (possibly unbounded) star-shaped polygon whose kernel contains $CH(Q)$. The boundary of $\overline{A}$ is formed by a set of polygonals, $\mathcal{POL}$, around $CH(Q)$ (in [4] it is used a similar idea). We will describe how to obtain $\mathcal{POL}$. First, we eliminate irrelevant points of P. Consider the following *dominance relation respect to $CH(Q)$*.

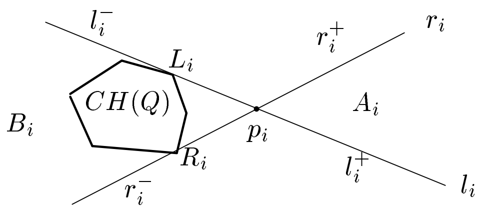


Figure 3: Regions A_i and B_i.

Definition 3. *p_i dominates p_j respect to $CH(Q)$ ($p_j \prec_Q p_i$), if $A_i \subset A_j$.*

The relation $\prec_Q$ is a partial order in P. A point p_j is minimal respect to $CH(Q)$ if there exists no i, $i \neq j$, such that $A_j \subset A_i$. It is clear that in the computation of A we only need the minimal points, the rest are irrelevant. The dominance relation is a generalization of the *vectorial dominance relation* [8]. The problem of the computation of minimal points can be formulated in the language of partial orders [12]. First at all, we introduce the notation and we order the points of P with respect to $CH(Q)$.

Notation: Let l_i (r_i) be the support line from p_i to $CH(Q)$ such that $CH(Q)$ is on the left (right) side going from p_i to $CH(Q)$. We compute the support lines, their angles and the tangency points L_i (R_i) of l_i (r_i) with $CH(Q)$. By l_i^+ or l_i^- we denote the half line given by l_i and the tangency point L_i depending on whether it contains or not the point p_i. Analogously for r_i^+ or r_i^- (see Figure 3).

Ordering points: We order the points of P according to the order of the half lines l_i^+, which are ordered according to: first, the order of L_i in $CH(Q)$ and second, the decreasing angle. If there are several points of P sharing a half line l_i^+, the order is given by the proximity to L_i. Relabeling p_i and r_i^+ we have the ordering $< (l_1^+, p_1, r_1^+), (l_2^+, p_2, r_2^+), \ldots, (l_n^+, p_n, r_n^+) >$, that corresponds to a clock-wise sweep line around $CH(Q)$. Let $L = \{p_1, p_2, \ldots, p_n\}$ be the ordered points of P. This process can be done in $O(n \log N)$ time.

Minimal points: A point p_i is not a minimal point if it dominates some point p_j. The procedure MINIMAL analyzes the points of P and deletes the no minimal points. Deciding if $p_j \prec_Q p_i$ requires constant time, the number of times the procedure analyzes a point is constant, so the running time of the procedure is $O(n)$. Note that the last points can dominate the first one.

MINIMAL
Input: list L (list of the ordered points of P)
Output: list MIN of minimal points
$MIN := L$, $i := 1$, $j := 2$
while $j \leq n$ **do**
  if $p_j \prec_Q p_i$ **then** find the first p_k in MIN such that $p_j \prec_Q p_k$, $k \leq i$; delete $\{p_k, \ldots, p_i\}$ of MIN
  $i := j$, $j := j + 1$
if $p_1 \prec_Q p_i$ **then** find the first p_k de MIN such that $p_1 \prec_Q p_k$, $k \leq i$; delete $\{p_k, \ldots, p_i\}$ of MIN.

Lemma 1. *The procedure MINIMAL obtains only all the minimal points of P respect $CH(Q)$.*

Proof. The procedure only deletes points that dominate other points. Suppose we have the closed chain obtained by the procedure. If a point p_i is minimal it does not dominate any other point and therefore the procedure does not delete it. If p_i is not minimal then there exist a minimal point p_j such that $p_j \prec_Q p_i$ with $i < j$. Suppose the procedure p_j does not eliminate p_i, then there exist a point p_k of the chain such that $i < k < j$ and $p_j \not\prec_Q p_k$. Therefore, p_k belongs to the region delimited by l_i^+, r_j^+ and l_j^+ (see Figure 4). Reasoning similarly, p_{i+1} belongs to this region and p_{i+1} will eliminate p_i. □

Polygonals: By $l_i^+ p_i$ and $r_i^+ p_i$ we denote the half lines l_i^+ and $r6_i^+$ from p_i. By t_{ij} we denote the intersection point of the half lines $l_i^+ p_i$ and $r_j^+ p_j$, and by $p_i t_{ij}$ or $t_{ij} p_j$ we denote the segments determined by their extreme points. The procedure POLYGONALS on MIN outputs $\mathcal{POL}$ which has at most $2n$ segments and half lines and it is the boundary between A and $\overline{A}$. The running time of the procedure is $O(n)$.

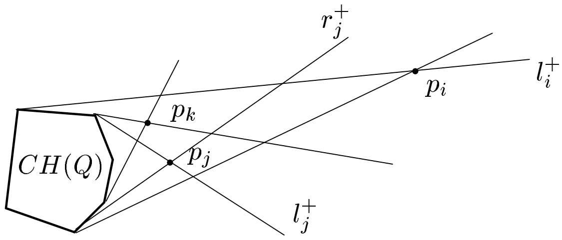


Figure 4: Minimal points.

POLYGONALS
Input: MIN (list of minimal points)
Output: $\mathcal{POL}$ (list of segments and half lines)
$\mathcal{POL}$:=$\{r_1^+p_1\}$, $i := 1$, $j := 2$
while $j \leq |MIN|$ **do**
  if there is intersection t_{ij} **then** add $\{p_it_{ij}, t_{ij}p_j\}$ to $\mathcal{POL}$, **else** add $\{l_i^+p_i, r_j^+p_j\}$ to $\mathcal{POL}$
  $i := j$, $j := j + 1$
if there is intersection t_{i1} **then** delete $\{r_1^+p_1\}$ of $\mathcal{POL}$, add $\{p_it_{i1}, t_{i1}p_1\}$ to $\mathcal{POL}$,
else add $\{l_i^+p_i\}$ to $\mathcal{POL}$.

2) Computing $\overline{B}$

The region B_i is delimited by the half lines l_i^- and r_i^- and part of $CH(Q)$. If $p_i \prec_Q p_j$ then $B_j \subset B_i$, and again, we only need the minimal points. Let p_i, p_{i+1} be consecutive points from MIM. We have the following cases:

i) p_i and p_{i+1} are linked by $\mathcal{POL}$. The region $B_i \cup B_{i+1}$ is delimited by the half line r_i^-, the segments in $CH(Q)$ between R_i and L_{i+1} and the half line l_{i+1}^-, (see Figure 5 a)).

ii) p_i and p_{i+1} are the final point and the initial point of two consecutive polygonals of $\mathcal{POL}$. In this case we have two different situations:

1. the half lines l_i^- and r_{i+1}^- are parallels and the region $B_i \cup B_{i+1}$ is as in i);
2. the half lines l_i^- and r_{i+1}^- intersect at a point denoted by p'_i, the region $B_i \cup B_{i+1}$ is as in i), but the wedge $\{l_i^-p'_i, r_{i+1}^-p'_i\}$ belongs to $\overline{B}$, as it is showed in Figure 5 b).

Note that if P and Q are not line separable then $\overline{B}$ is formed only by disjoint wedge regions. Looking over $\mathcal{POL}$ in $O(n)$ time we can obtain the at most n disjoint wedges of $\overline{B}$. Let L' be the list of ordered points p'_i corresponding to the wedge regions $\{l_i^-p'_i, r_{i+1}^-p'_i\}$ of $\overline{B}$. We denote by $l_i'^+$ the half line r_{i+1}^- and by $r_i'^+$ the half line l_i^-.

3) Computing $\overline{A} \cap \overline{B}$

First, note that a wedge of $\overline{B}$ can be totally included in the region A, in such case the wedge is *useless*. This situation happens if the point p'_i defining a wedge of $\overline{B}$ dominates a point p_k of $\mathcal{POL}$.

Useless wedges: In $O(n \log n)$ time we order $MIN \cup L'$ in the same way we did it for the minimal point computation. The procedure USELESS WEDGES deletes the points of L' dominated by some point of MIN in $O(n)$ time.

USELESS WEDGES
Input: list of ordered points of $MIN \cup L'$
Output: L' without points dominated by points of MIN
while there are points p'_i of L' **do**
  find the first p_k of MIN following p'_i, $(i < k)$, **if** $p_k \prec_Q p'_i$ **then** delete p'_i from L'.

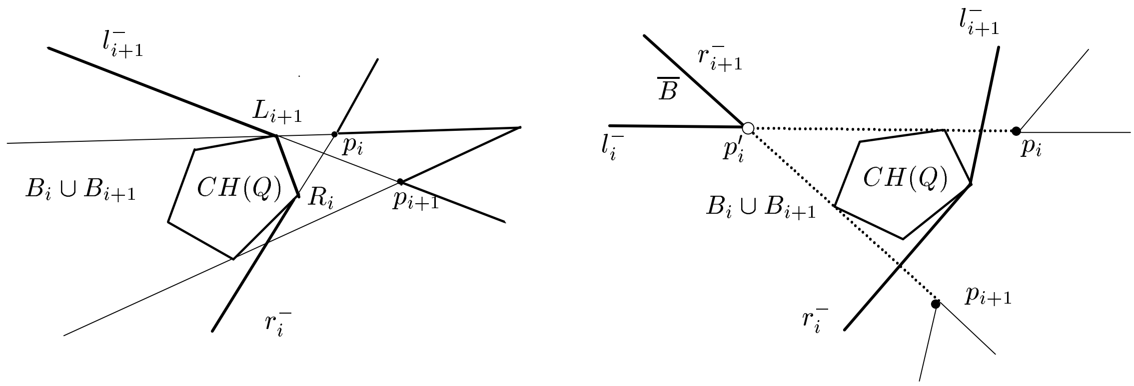


Figure 5: a) Regions of B, b) regions of $\overline{B}$.

Second, the half lines l'^+_i and r'^+_i of a useful wedge of $\overline{B}$ intersect $\mathcal{POL}$ in at most one point (l'^+_i is between some l^+_j and l^+_{j+1}) (analogously for r'^+_i), or it contains some segment or half line of $\mathcal{POL}$. Therefore, the number of intersections between $\mathcal{POL}$ and the half lines of wedges of $\overline{B}$ is less or equal to $2n$.

Regions of $\overline{A} \cap \overline{B}$: The regions of $\overline{A} \cap \overline{B}$, denoted by F_i, are the intersections between the useful wedges of $\overline{B}$ and the (possibly unbounded) star-shaped polygon $\overline{A}$. Since a point p'_i of L' is within $\overline{A}$, the region F_i is delimited by the half lines of the useful wedge and the part of $\mathcal{POL}$ between the half lines. We say that the region F_i is a *fan* since it is still a (possibly unbounded) star-shaped polygon whose kernel contains the point p'_i, *the apex of the fan*, and the angle of the useful wedge at p'_i is less than π (see Figures 6 and 7). The vertices of a fan are reflex vertices if they are points of P, *the tips of the fan*, otherwise they are convex vertices. No two reflex vertices are adjacent. The procedure REGIONS outputs the fans of $\overline{A} \cap \overline{B}$ in $O(n)$ time. The number of segments and half lines of all the fans is at most $4n$.

REGIONS
Input: L', $\mathcal{POL}$
Output: segments and half lines of $\overline{A} \cap \overline{B}$
while there are points p'_i of L' **do**
  compute F_i: region delimited by the wedge $\{l'^+_i p'_i, r'^+_i p'_i\}$ and the part of $\mathcal{POL}$ between l'^+_i and r'^+_i

From all the above discussion we have the following theorem.

Theorem 1. *Let P and Q be two disjoint sets of n and m points in the plane respectively. The region of vertices of the separating wedges can be computed in $O(N \log N)$ time.*

2.1.1 Wedges with maximum and minimum angle

Suppose we have computed the regions of vertices of separating wedges and we want to determine the wedges with maximum and minimum angle.

Lemma 2. *In a fan F_i, the vertex of the separating wedge with maximum angle is the apex of the fan. If the fan is bounded, the vertex of the separating wedge with minimum angle is one of the convex vertices,* the tips of the fan*. If the fan is unbounded the minimum angle is zero.*

Proof. Let p be an interior point of the fan F_i. The support lines from p to $CH(Q)$ intersect the half lines l'^+_i and r'^+_i at points u_1 and u_2, respectively (see Figure 7). If the vertex of a separating wedge moves from p till u_1 and from u_1 till p'_i the angle of the separating wedge increase continuously and analogously, when the vertex moves from p till u_2 and from u_2 till p'_i. Therefore, the vertex of the separating wedge with maximum angle is p'_i, the apex of the fan.

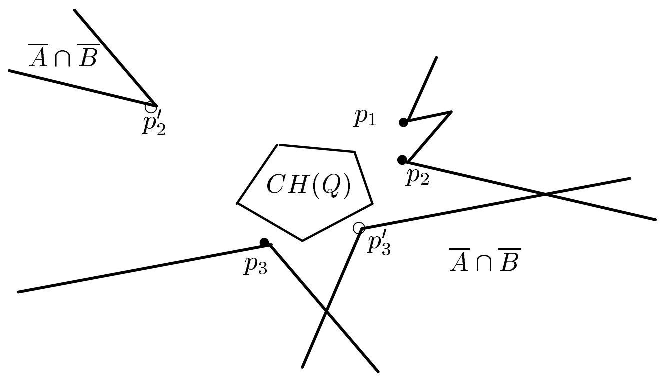


Figure 6: Regions of $\overline{A} \cap \overline{B}$.

If the fan is bounded, the support lines from p intersect the part of $\mathcal{POL}$ limiting the fan region at points v_1 and v_2. Moving the vertex of the separating wedge from p till v_1 and from v_1 through the segment of the $\mathcal{POL}$ that contains v_1 till the right extreme of the segment, the angle of the separating wedge decreases continuously. Analogously, when the vertex moves from p till v_2 and from v_2 till the left extreme of the segment that contain v_2. Therefore, the vertex of the separating wedge with the minimum angle is in some tip of the fan. If the fan is unbounded, the minimum angle is zero for a separating wedge with the vertex at infinity point. The total number of tips of fans is at most n. Doing a tour in $\mathcal{POL}$ we determine the wedges with maximum and minimum angle in $O(n)$ time. $\square$

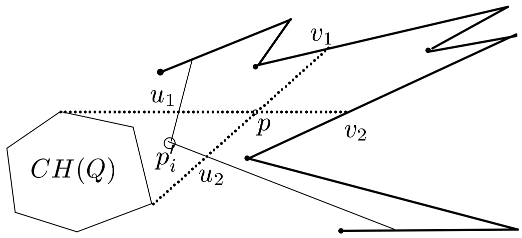


Figure 7: Maximum and minimum angle.

Proposition 1. *Let P and Q be two disjoint wedge separable sets of n and m points in the plane respectively. Once we have pre-computed the vertices of separating wedges, the wedges with maximum and minimum angle can be found in $O(N)$ time.*

2.2 Separating segments by wedges

Let $P = \{p_1p_1', p_2p_2', \ldots, p_np_n'\}$ and $Q = \{q_1q_1', q_2q_2', \ldots, q_mq_m'\}$ be two disjoint segments sets. If there is a wedge separating Q from P, the wedge contains the convex hull of the segments of Q (see Figure 8 a)). In $O(N \log N)$ time we compute the convex hull of each segment set, determine if any of them is monochromatic, i.e., it does not intersect segments or does not contain extremes of segments of the other set, otherwise they are not wedge separable.

Suppose that the segment sets are not line separable and that $CH(Q)$ is monochromatic. We want to compute the region of vertices of wedges separating Q from P. First, we consider the restriction produced by a segment $s_i = p_ip_i'$ of P. The support lines between s_i and $CH(Q)$ define the regions A_i and B_i that do not contain vertices of separating wedges, otherwise the wedge contains totally or partially the segment s_i (see Figure 8 b)). The region of vertices of separating wedges is $\overline{A} \cap \overline{B}$, where $A = \cup A_i$ and $B = \cup B_i$.

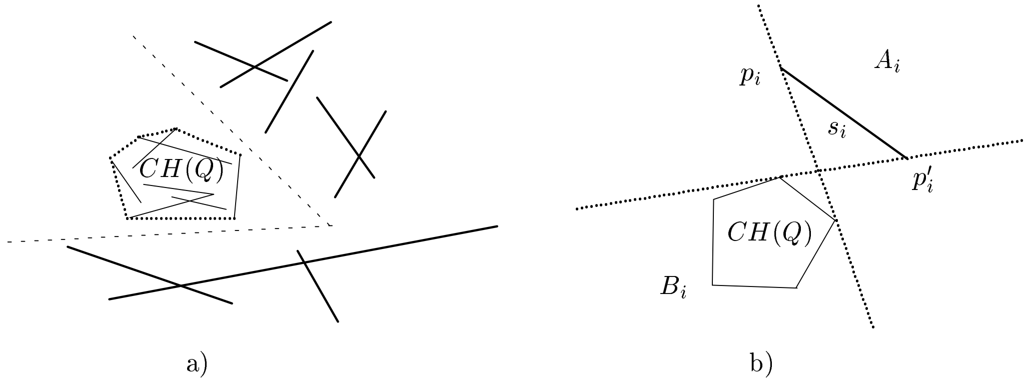


Figure 8: a) Separating wedge, b) restriction of a segment.

1) **Computing $\overline{A}$**

The region $\overline{A}$, as we are going to see, is again a (possibly unbounded) star-shaped polygon whose kernel contains $CH(Q)$. The boundary of $\overline{A}$ is formed by a set of polygonals, *final polygonals* or POL for short, around $CH(Q)$. We will describe how to obtain the final polygonals.

Ordering segments: We order the segments of P with two orders respect $CH(Q)$: clock-wise and counter clock-wise order. For the clock-wise order we compute the lines supporting $CH(Q)$ from all the end points of segments of P. Let l_i^+ $({l'}_i^+)$ be the half line tangent to $CH(Q)$ going through p_i $({p'}_i)$ such that $CH(Q)$ is on the left side going from p_i $({p'}_i)$ to $CH(Q)$ and beginning at the tangency point (see Figure 9 a)). We order the half lines tangents in angular clock-wise way and we order the segments according to the first extreme that appear in the order given by the half line tangents. If there are several segments with the first extremes sharing the same half line, the order is given by the proximity of the extreme to the tangency point of the half line, if some of these segments have the same first extremes, the order is given by the order of the second extreme. Once we have ordered the segments we denote by p_i the first extreme of the segment and by ${p'}_i$ the second one. Let $s_1 = p_1p'_1$ be the first segment of P in this order. If a segment $s_i = p_ip'_i$ intersects l_1^+ at a point $p_i^{''}$ we divide it in two segments $p_ip_i^{''}$ and $p_i^{''}p'_i$ and delete the segment s_i. The number of segments increase in at most $n-1$. Reordering the segments with the new extremes according to the criteria above, we are sure that doing a clock-wise angular sweep beginning at l_1^+ we find all the segments after just one round.

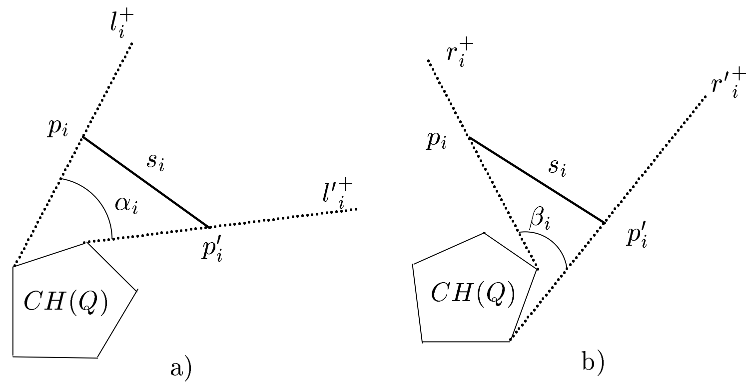


Figure 9: a) Clock-wise angular interval, b) counter clock-wise angular interval.

For the counter clock-wise ordering we proceed in a similar way using the order of r_i^+ $({r'}_i^+)$, half line tangents to $CH(Q)$ from the point p_i $({p'}_i)$ and beginning at the tangency point that has $CH(Q)$ on his right going from p_i (p'_i) to the tangency point (see Figure 9 b)). Again, once we have the counter clock-wise ordering, we denote by p'_i the first extreme of a segment $s_i = p'_i p_i$ and by p_i the second extreme. We can take $s_1 = p'_1 p_1$ as the first segment of P and use r'^{+}_1 as the starting half line and proceed as above. All the ordering processes require $O(n \log N)$ time.

Final polygonals: Analogously as in the case of points, the polygonals of $\mathcal{POL}$ are formed by segments and half lines. From a point of any polygonal of $\mathcal{POL}$ we *see* $CH(Q)$ without any obstacle, i.e., the region determined by the half line tangents to $CH(Q)$ from the point of the polygonal and the part of $CH(Q)$ between the tangency points does not contain or intersect segments of P.

Starting with s_1, each segment has an angular projection on the interval $[0, 2\pi)$ sweeping clock-wise (counter clock-wise) with half line tangents to $CH(Q)$: each segment s_i has a clock-wise angular interval α_i (see Figure 9 a)) and a counter clock-wise angular interval β_i (see Figure 9 b)). α_i (β_i) is the clock-wise (counter clock-wise) angle defined by the half lines l^{+}_i and l'^{+}_i (r'^{+}_i and r^{+}_i). With the above ordering of segments, we are sure that all the angular intervals are within $[0, 2\pi)$. Now we describe how to compute $\mathcal{POL}$:

1) compute the set of polygonals of segments that can be seen from $CH(Q)$ (without obstacles) doing a clock-wise rotating sweep with a half line tangent to $CH(Q)$ such that $CH(Q)$ is on the left. We call this set *the clock-wise polygonals* (see Figure 10 a));

2) compute the set of polygonals of segments that can be seen from $CH(Q)$ (without obstacles) doing a counter clock-wise rotating sweep with a half line tangent to $CH(Q)$ such that $CH(Q)$ is on the right. We call this set *the counter clock-wise polygonals* (see Figure 10 b));

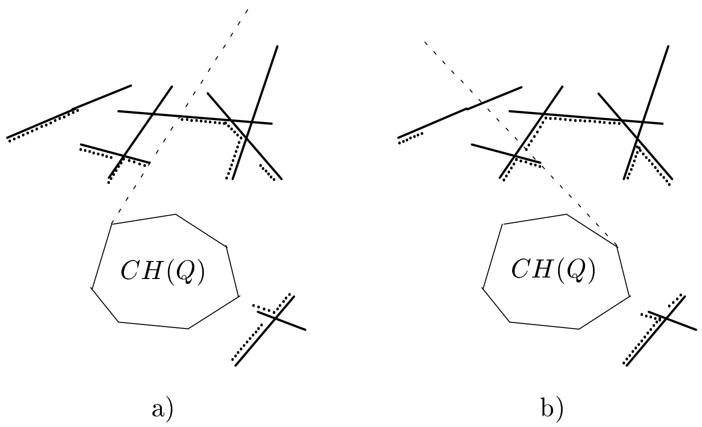


Figure 10: a) Clock-wise polygonals, b) counter clock-wise polygonals.

3) compute the set of polygonals corresponding to the extremes of segments of P. We call this set *the extreme polygonals* (see Figure 11 a)). It can be computed as the polygonal constructed in the case of points;

4) from the three sets of polygonals construct the set of final polygonals or $\mathcal{POL}$ (see Figure 11 b)).

The clock-wise polygonals are a variation of the *lower envelope of a segment set*, we denote it by *clock-wise angular lower envelope*. Modifying the Hershberger algorithm for the lower envelope of a set of x-monotone Jordan arcs with at most s intersections between any pair of arcs [6, 15] in such a way that for each segment point we define the x-coordinate as the angle of the tangent line to $CH(Q)$ from the point that has $CH(Q)$ on the left as going from the point to $CH(Q)$ and we define $f(x)$ as the geodesic distance between the point and the tangency point with $CH(Q)$ of the tangent line that pass through the first extreme of the segment. The segments so defined are x monotone and with at most one intersection between any pair of them. If there are segments with two common points, *partially coincident segments*, in $O(n \log n)$ time we can construct an equivalent segment set without partially coincident segments. Therefore, the clock-wise polygonals can be computed in $O(N \log N)$ time.

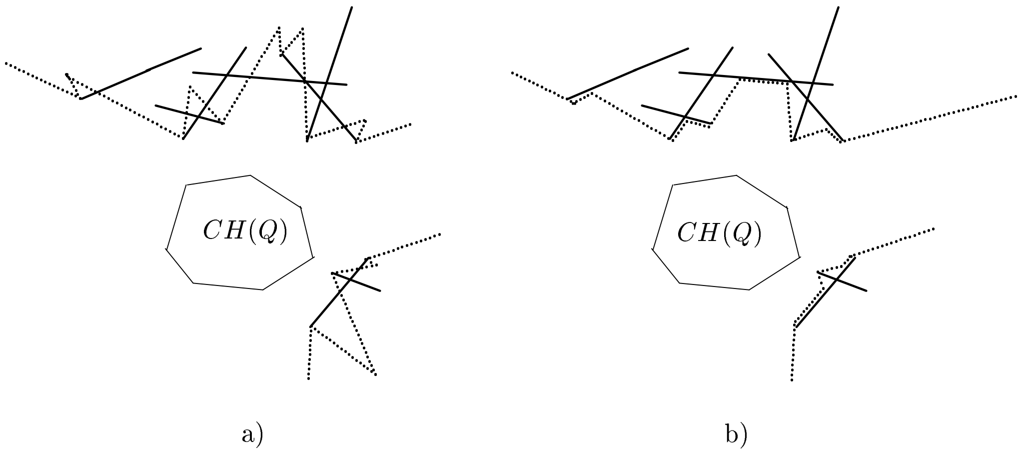


Figure 11: a) Extreme polygonals, b) final polygonals.

Analogously, we can compute the counter clock-wise polygonals beginning from the extreme p_1' of s_1 and using the counter clock-wise ordering. The extreme polygonals can be computed in $O(N)$ time using the order of the segment extremes as we did it in the wedge separability of point sets. The extreme polygonals are formed by at most $4n$ segments and half lines.

For the construction of the polygonals of $\mathcal{POL}$, we merge the above three sets of polygonals taking into account that the number of transition points of the clock-wise and counter clock-wise polygonals is $O(n\alpha(n))$, where $\alpha(n)$ is the inverse of the Ackermann function. Making a clock-wise rotating angular sweeping advancing in discrete steps over the angular intervals determined by the $O(n\alpha(n))$ transition points of the three sets of polygonals, we can determine in constant time which of the three polygonals we take in each angular interval between transition points. The total running time of this part is $O(n\log n)+O(n\alpha(n)) = O(n\log n)$.

The total time needed in the construction of $\mathcal{POL}$ (the boundary between A and $\overline{A}$) is $O(n\log N)$. Note that the number of segments and half lines of the $\mathcal{POL}$ is bounded by $O(n\alpha(n))$, and by construction, $\mathcal{POL}$ is clock-wise and counter clock-wise angularly monotone.

2) **Computing $\overline{B}$**

To compute $\overline{B}$ we follow the same criteria as in the case of the wedge separability of point sets. Two connected and consecutive parts (polygonals) of the final polygonal give a wedge region of $\overline{B}$ defined by the half lines l_i^- from the end of the first part and r_{i+1}^- from the beginning of the second part if the half lines l_i^+ and r_{i+1}^+ are not parallel. We denote the intersection point by $p_i^{'''}$. Looking over the final polygonal we determine all the wedge regions of $\overline{B}$ in $O(n\alpha(n))$ time. Note that the number of this regions is at most n.

3) **Computing $\overline{A} \cap \overline{B}$**

A wedge region of $\overline{B}$ is *useless* if the point $p_i^{'''}$ is in A, since the wedge region has not intersection with $\overline{A}$. To determine all the useless wedge regions of $\overline{B}$ we analyze the points $p_i^{'''}$ checking if they lie in A or in $\overline{A}$ with an $O(\log n)$ cost for each point. Once we have eliminated all the useless wedge regions of $\overline{B}$, we compute the part of the final polygonals between the half lines that define the useful wedge regions. By the clock-wise and counter clock-wise angular monotony, each of this half lines intersect the final polygonal in only one point (or it contains a segment or a half line of the final polygonals). In $O(N\log N)$ time we compute $\overline{A} \cap \overline{B}$. Note that these regions are (possibly unbounded) star-shaped polygons or fans whose kernel contains the point $p_i^{'''}$. As a consequence of all the above discussion we have the following theorem.

Theorem 2. *Let P and Q be two disjoint sets of n and m segments in the plane respectively. To decide if they are wedge separable computing the region of vertices of the possible wedges can be done in $O(N\log N)$ time.*

2.2.1 Wedges with maximum and minimum angle

The wedge with the maximum angle is in the apex of the fan regions of $\overline{A} \cap \overline{B}$. It can be determined computing in constant time the angle of the wedge with vertex in p_i''' for each of the at most n fan regions.

If there is an unbounded fan region the minimum angle is zero. Otherwise, the wedge with minimum angle is in the convex extremes of the fan, *the tips*. An upper bound of the total number of tips is $O(n\alpha(n))$ and for each tip we compute his angle in constant time (see Figure 12). Checking all the tips we compute the wedge with minimum angle in $O(N\alpha(N)$ time.

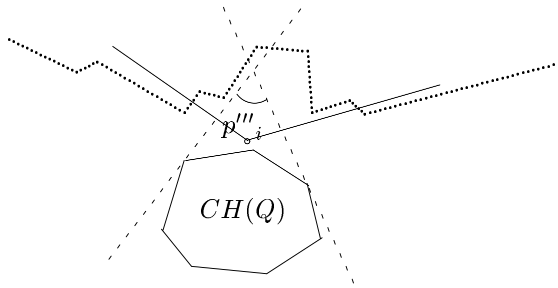


Figure 12: Wedges with maximum and minimum angle.

Proposition 2. *Let P and Q be two disjoint wedge separable sets of n and m segments in the plane respectively. If we have pre-computed the fan regions of vertices of wedges, the wedges with maximum and minimum angle can be found in $O(n\alpha(n))$ time.*

2.2.2 An application to circular separability

In [2] it is showed that for two sets of line segments P and Q with a total of n segments meeting only at their endpoints, it is possible to compute in $O(n \log n)$ time and $O(n)$ space all the largest circles separating P and Q. For the general case they showed the following theorem.

Theorem 3 (*Boissonnat et al.*). *For two sets of line segments P and Q containing a total of n segments, it is possible to compute in $O(n\alpha(n) \log^2 n)$ deterministic time or in $O(n\alpha(n) \log n)$ randomized time and $O(n\alpha(n)$ space all locally largest circles separating P and Q.*

As pointed out in [2]: *to prove the theorem it is sufficient to take into consideration separation of the external cell of the arrangement of line segments of set P, and, containing it, a single cell of the arrangement of line segments of Q.*

It is clear that if there exist a circle separating Q from P, a largest circle C has to contain $CH(Q)$ and has to be tangent to segments of P in some points. The tangency points of C with segments of P satisfy the property that this points see $CH(Q)$: we say that a point p *see* $CH(Q)$ if the region defined for the half line tangents from p to $CH(Q)$ and the part of $CH(Q)$ between the half line tangents does not contain any obstacle. Then, any circle separating Q from P separates also $CH(Q)$ from the set of segments of the final polygonals obtained in this section for the wedge separability of Q from P. The final polygonals are formed by $O(n\alpha(n))$ segments meeting only at their endpoints and we have computed the final polygonals in $O(n \log n)$ time. Then applying the above theorem to the n segments of Q and $O(n\alpha(n))$ segments of the final polygonals, it is possible to compute all the largest circles separating Q from P in time $O(n\alpha(n) \log(n\alpha(n)) = O(n\alpha(n) \log n)$.

Theorem 4. *For two sets of line segments P and Q containing a total of n segments, it is possible to compute in $O(n\alpha(n) \log n)$ deterministic time and $O(n\alpha(n))$ space all locally largest circles separating P and Q.*

2.3 Separating polygons by wedges

The wedge separability of polygons depends on the total number of segments of the polygons and the problem can be reduced to the case of separability of segments. Let P be a set of a polygons, each one with n_i segments ($n_1 + \ldots + n_a = n$) and Q a set of b polygons, each one with m_i segments ($m_1 + \ldots + m_b = m$). Let $CH(Q)$ be the convex hull of the polygons of Q. In time $O(n \log m)$ we can check if all the polygons of P are exteriors to $CH(Q)$, i.e., there are not segments of the polygons of P that intersect $CH(Q)$ and there are not polygons of P contained in $CH(Q)$. Otherwise Q is not wedge separable from P. If all the polygons of P are exterior to $CH(Q)$, consider the set of segments of the polygons of P and run the above segment algorithm.

Corollary 1. *Let P and Q be two disjoint polygon sets of the plane with n and m total segments respectively. To decide if they are wedge separable computing the regions of vertices of wedges and the wedges with maximum and minimum angle can be done in $O(N \log N)$ time.*

2.4 Separating circles by wedges

Let P and Q two disjoint circle sets with n and m circles classified as blue and red circles respectively. In $O(m \log m)$ time we compute $CH(Q)$ formed by at most $2m-1$ circular arcs and $2m-1$ segments [3,13] and in $O(n \log m)$ time we check that the circles of P do not intersect and are exterior to $CH(Q)$. If it is so, in $O(n \log N)$ time we compute the interior tangents between each circle of P and $CH(Q)$, delete possible circle duplicities of P and determine the circular arc of each circle of P that can take part in the computation of the vertices of separable wedges (see Figure 13 a)). Let P' the set of this circulars arcs. As in the case of the segment sets, the interior tangents between each circular arc s_i and $CH(Q)$ define two regions A_i and B_i that do not contain vertices of wedges, otherwise the wedge contains total or partially the circular arc s_i. The region of vertices of wedges is $\overline{A} \cap \overline{B}$ where $A = \cup A_i$ and $B = \cup B_i$. We follow the same process as above.

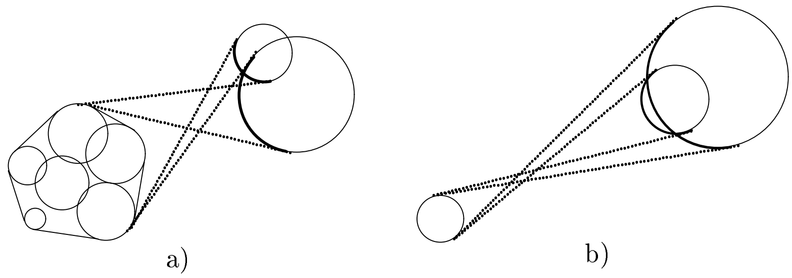


Figure 13: Circles.

1) **Computing $\overline{A}$**

We compute the *final polygonals* or $\mathcal{POL}$ formed by circular arcs, segments and half lines that, as we are going to see, separate A from $\overline{A}$ (see Figure 14).

Ordering circular arcs: We order the circular arcs $s_i = p_i p'_i$ of P' (p_i and p'_i are the extremes) with two orders with respect to $CH(Q)$: clock-wise and counter clock-wise. We proceed as for the segment sets. For the clock-wise ordering we use the tangents to $CH(Q)$ from the extremes of the circular arcs of P'. Let l_i^+ (l'^+_i) be the half line tangent to $CH(Q)$ containing the point p_i (p'_i) and beginning at the tangency point that has $CH(Q)$ on its left going from p_i (p'_i) to the tangency point. We order the half lines in angular clock-wise and we order the circular arcs according to the first extreme that appear in the order given by the half line tangents. If there are several circular arcs with the first extreme sharing the same half line, the order is given by the proximity of the first extreme of the circular arc to the tangency point of the half line with $CH(Q)$. Once we have ordered the circular arcs we denote by p_i the first extreme of the circular arc and by p'_i the second extreme. Let $s_1 = p_1 p'_1$ be the first circular arc of P'. If a circular arc $s_i = p_i p'_i$

intersects l_1^+ in a point $p_i^{''}$ we divide it in two circular arcs $p_ip_i^{''}$ and $p_i^{''}p_i'$ and delete s_i. The number of circular arcs increases in at most $n-1$. Let P' be the set of this circulars arcs. Reordering P' according to the above criteria we are sure that doing a clock-wise angular sweep starting in l_1^+ we find all the circular arcs after just one round.

For the counter clock-wise ordering we proceed in a similar way using the order of r_i^+ $({r'}_i^+)$, half line tangent to $CH(Q)$ containing the point p_i $({p'}_i)$ and beginning at the tangency point that has $CH(Q)$ on his right going from p_i $({p'}_i)$ to the tangency point. Again, once we have the counter clock-wise ordering we denote by ${p'}_i$ the first extreme of a circular arc $s_i = p_ip_i'$ and by p_i the second extreme. We can take $s_1 = p_1'p_1$ as the first circular arc of P', use ${r'}_1^+$ as the starting half line, and proceed as above. All the ordering processes require $O(n \log N)$ time.

Final polygonals: The polygonals of $\mathcal{POL}$ are formed by circular arcs, segments and half lines. From any point of one polygonal of $\mathcal{POL}$ we *see* $CH(Q)$ without any obstacle, i.e., the region determined by the half line tangents to $CH(Q)$ from the point of the polygonal and the part of $CH(Q)$ between the tangency points does not contain circular arcs or extremes of circular arcs of P'.

As in the case of segments, the circular arcs define a clock-wise (counter clock-wise) angular interval obtained doing a clock-wise (counter clock-wise) rotational sweep with the half line tangents to $CH(Q)$. The circular arcs are angular-monotone and we can apply the same process described for the segment sets.

Observation *Two circular arcs of P' can intersect in two points (see Figure 13 b)).*

The difference comes from the observation: the computation of the clock-wise (counter clock-wise) angular lower envelope with the modification of the Hershberger algorithm [6,15] requires $O(\lambda_3(n) \log n) = O(n\alpha(n) \log n)$ time.

Nevertheless, the complexity of the clock-wise (counter clock-wise) angular lower envelope of blue circular arcs is linear since it is less or equal than the combinatorial complexity of a *single face* in a arrangement of n closed Jordan curves (blue circles) in the plane, which is $\lambda_2(n) = 2n - 1$ (see theorem 5.7 of [15]). Analogously for the counter clock-wise angular lower envelope and for the merging of both envelopes. Taking into account this complexity, we compute the clock-wise angular lower envelope using the algorithm given by the corollary 6.2 of [15], that consists of making a partition of P' in two subsets, each of size at most $\lceil \frac{n}{2} \rceil$, compute the clock-wise angular lower envelope of each of them recursively and then merge these subenvelopes to obtain the overall envelope. For merging the two clock-wise angular lower envelopes we sweep clock-wise angularly a half line tangent to $CH(Q)$ advancing in discrete steps over the clock-wise angular intervals determined by the transition points of the subenvelopes and spending $O(1)$ time in each step. The merge step requires $O(\lambda_2(n))$ time. If $T(n)$ is the maximum running time of the algorithm, we obtain the recurrence $T(n) = 2T(\frac{n}{2}) + O(\lambda_2(n))$. The solution of the recurrence is $O(n \log n)$.

In the same way, we compute the counter clock-wise angular lower envelope. The merge of the two envelopes, (*circle polygonals*), is obtained as the above merging in $O(\lambda_2(n))$ time, which is the number of transition points. With the order of the extremes of the circular arcs we compute the *extreme polygonals* formed by segments and half lines in $O(n)$ time. To get the final polygonals or POL we merge the circle polygonals and the extreme polygonals in $O(n)$ time making a clock-wise angular sweep with a half line tangent to $CH(Q)$ advancing in discrete steps over the clock-wise angular intervals of the circular arcs and segments. The complexity of the final polygonals is $O(n)$. Note that $\overline{A}$ is a (possibly unbounded) star-shaped polygon (with also circular arcs as edges) whose kernel contains $CH(Q)$.

2) **Computing $\overline{B}$**

The region $\overline{B}$ is computed as the analogous region for the case of the segment sets. Two connected and consecutive polygonals of the final polygonals give a wedge region of $\overline{B}$ defined by the half lines l_i^- from the end of the first polygonal and r_{i+1}^- from the beginning of the second polygonal if l_i^+ and r_{i+1}^+ are not parallel. We denote the intersection point by $p_i^{'''}$. Using the final polygonals we determine all the at most n wedge regions of $\overline{B}$ in $O(n)$ time.

3) **Computing $\overline{A} \cap \overline{B}$**

A wedge region of $\overline{B}$ is useless if the point $p_i^{'''}$ lies in A. We analyze the points $p_i^{'''}$ checking if they lie in A or in $\overline{A}$ with an $O(\log n)$ cost for each point. Once we have eliminated all the useless wedge regions of $\overline{B}$, we compute the part of the final polygonals between the half lines that define the useful wedge regions.

Each of this half lines intersect the final polygonals in at most one point (or it contains a half line of the final polygonals) by the clock-wise and counter clock-wise angular monotony. In $O(N \log N)$ time we compute the regions $\overline{A} \cap \overline{B}$. Note that these regions are still (possibly unbounded) star-shaped polygons (with also circular arcs as edges) or fans whose kernel contains the point p_i''' (see Figure 14). As a consequence of the above discussion we have the following theorem.

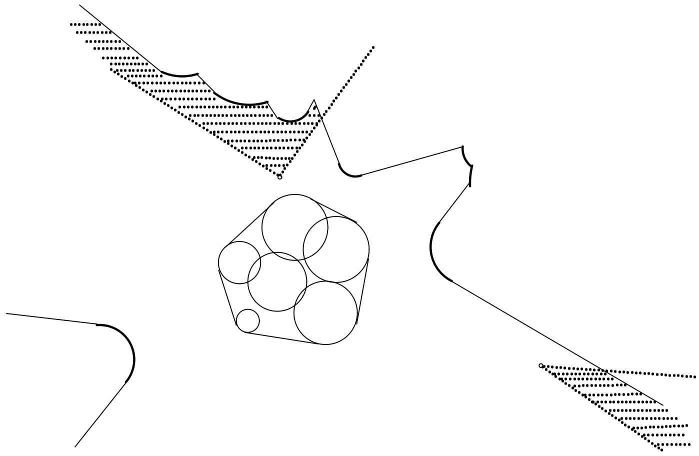


Figure 14: Regions of wedge vertices for circles.

Theorem 5. *Let P and Q two disjoint sets of n and m circles in the plane respectively. To decide if they are wedge separable computing the region of vertices of wedges can be done in $O(N \log N)$ time.*

2.4.1 Wedges with maximum and minimum angle

The wedge with the maximum angle is in the apex of the fan regions of $\overline{A} \cap \overline{B}$. It can be determined computing in constant time the angle of the wedge with vertex in p_i''' for each of the at most n fan regions. If there is an unbounded fan region the minimum angle is zero. Otherwise, the wedge with minimum angle is in the convex extremes of the fan, *the tips*. The number of tips is $O(n)$ and for each tip we compute its angle in constant time (see Figure 14).

Proposition 3. *Let P and Q be two disjoint wedge separable sets of n and m circles in the plane respectively. The wedges with maximum and minimum angles can be found in $O(N)$ time if we have pre-computed the fan regions of vertices of wedges.*

3 Strip separability

We study now the problem of the separability of two disjoint object sets in the plane by a strip. Suppose that P and Q are the sets of *blue* and *red* objects respectively.

It is clear that if P and Q are strip separable they are wedge separable, just move a little bit one of the parallel lines that defines the strip. We can interpret a strip as a wedge with vertex at an infinity point. The difference is that before we looked for possible wedge vertices (two parameters) and now it is enough to determine the slope of the strip (one parameter). The slope of the strip is the slope of the parallel lines that defines the strip, where the slope is given by the clock-wise angle formed by one of the parallel lines with the horizontal and included in the interval $[0, \pi)$.

If P and Q are line separable they are strip separable and the slopes of the strips are between the slopes of the interior tangent lines of $CH(P)$ and $CH(Q)$. A necessary condition for the strip separability is that the convex hull of one of the sets is monochromatic. In $O(N \log N)$ time we compute $CH(P)$, $CH(Q)$ and we determine if there is a monochromatic set. We admit that the parallel lines of the strip can *lean on* the objects of P and Q. The solution of the problem is determine the slope intervals of the possible separating strips. Next, we study the problem of the strip separability for different kind of objects.

3.1 Separating points by strips

Let $P = \{p_1, p_2, \ldots, p_n\}$ and $Q = \{q_1, q_2, \ldots, q_m\}$ two disjoint point sets in the plane, the *blue* and *red* points respectively. Suppose that they are not line separable and $CH(Q)$ is monochromatic. We want to determine the slope intervals of the strips separating Q from P. First, note the restriction of a point p_i: the half line tangents l_i^+ and r_i^+ from p_i to $CH(Q)$ have counter clock-wise angles α_{l_i} and α_{r_i} with the horizontal. It is clear that the interval $(\alpha_{l_i}, \alpha_{r_i})$ contains no solutions, otherwise the strip would contain the point p_i. We denote this interval by *negative interval* of p_i (see Figure 15).

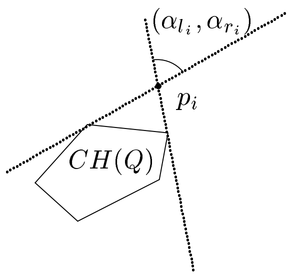


Figure 15: Negative interval.

Moreover, if $p_i \prec_Q p_j$, the negative interval of p_j is contained in the negative interval of p_i. Therefore, the final polygonals constructed in the above section correspond to the points of P that can take part in the computation of the strips. In fact, we only need the points $\{p_i, p_k\}$ initial and final of each polygonal of the final polygonals, giving the negative interval $(\alpha_{l_k}, \alpha_{r_i})$, ($CH(Q)$ can not *go out* between those points). Therefore, using the final polygonals and in $O(n)$ time, we obtain the at most n disjoint negative intervals within $[0, 2\pi]$. In $O(n)$ time we reduce the negative intervals in $[\pi, 2\pi]$ to intervals in $[0, \pi)$ and we compute the union. The complement of this union with respect to $[0, \pi)$ are the slope intervals of the separating strips, *strip intervals* for short. If the complement is the empty set, the sets are not strip separable. Note that the negative intervals are open and therefore the strip intervals are closed or points. From the above discussion we have the following proposition.

Proposition 4. *Let P and Q be two disjoint sets of n and m points in the plane respectively. To decide if they are strip separable computing the strip intervals can be done in $O(N \log N)$ time.*

A natural problem in the case that the sets are strip separable is to determine the narrowest and the widest strip. Suppose that Q is strip separable from P and we have computed the at most n strip intervals.

3.1.1 The narrowest strip

The narrowest strip is defined by two parallel lines tangent to $CH(Q)$ in antipodal points and it is clear that its computation depends only on $CH(Q)$ and on the strip intervals. Note that it is a similar problem to the computation of the width of a convex polygon, i.e., the small distance between parallel support lines of the polygon. In our case we compute the width of $CH(Q)$ restricted to the strip intervals.

We generate in $O(m)$ time all the antipodal points of $CH(Q)$ in the counter clock-wise order [14,17]. For each antipodal pair of points we compute in constant time the interval (or two intervals) in $[0, \pi)$ of common slopes of the support lines in these points. In the intersection of this interval with the strip intervals, the variation of the distance between support lines is monotone or upwards unimodal, therefore, the minimum distance corresponds to the extremes of the intersection. In constant time we compute the extremes, the minimum width of the strips corresponding to a pair of antipodal points and update the minimum. Note that it is possible to have strips with the minimum width in different antipodal points.

Proposition 5. *Let P and Q be two disjoint strip separable sets of n and m points in the plane respectively. If we have pre-computed the strip intervals, the narrowest strip can be found in $O(N)$ time.*

3.1.2 The widest strip

We exclude here the situation in which P and Q are line separable since in this case the widest strip is infinite. The widest strip only depends on the points of P in the final polygonals. We consider the problem in the dual plane by doing the transformation:

$$L : y = 2ax - b \longleftrightarrow p : (a, b), \qquad D(L) = p, \;\; D(p) = L.$$

We transform the points of $CH(Q)$ and P such that $D(CH(Q))$ are red lines and $D(P)$ are blue lines. Previous to transformation, we make a change of coordinates for simplifying the situation in the dual plane. If in the primal plane all the strip intervals are points, in $O(n)$ time we compute the widest strip. Otherwise we choose a strip interval $[\alpha_1, \alpha_2]$, $\alpha_1 \neq \alpha_2$ and we make a change of coordinates in $ON \log n)$ time taking as vertical a line with slope within (α_1, α_2) and as origin a point such that the points of $P \cup Q$ have different abscissa. In this way, the blue lines in the dual plane have different slopes within the slope interval (a_{q_1}, a_{q_m}), where q_1 and q_m are the red points with minimum and maximum abscissa respectively (see Figure 16).

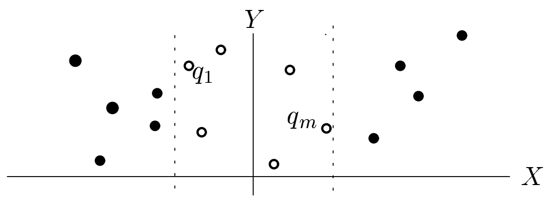


Figure 16: Changing coordinates.

We order the lines according to the increasing slope (counter clock-wise) departing from the vertical (there are not vertical lines). The situation in the dual plane is as show in Figure 17. The convex hull of $D(CH(Q))$ determine two regions: a upper region U and a lower region L, limited by E_U and E_L, respectively. All the red lines lie between E_U and E_V, but no any blue line, since in the primal plane there are not blue points inside the red convex hull. A blue line intersects E_U in one point and intersect E_L in other point, determining a *null vertical slab* defined by the two parallel vertical lines that pass through the intersection points of the blue line with E_U and E_L (see Figure 17).

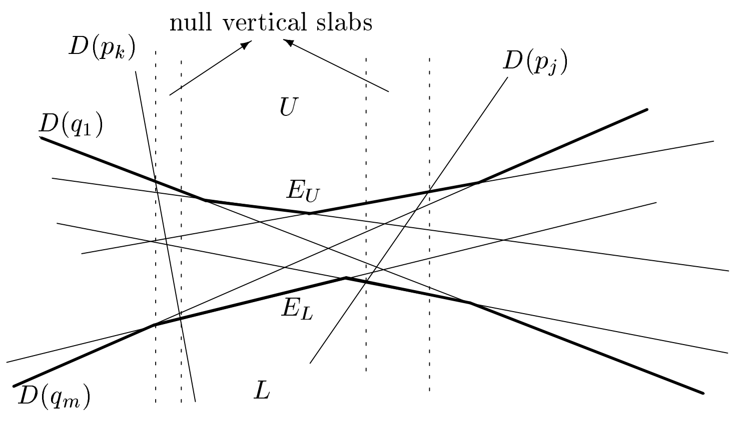


Figure 17: Dual arrangement.

In arrangement produced by the lines in the dual plane we consider the *upper cells* (in U) and the *lower cells* (in L) that are defined as the convex regions limited by segments of red lines, segments of blue lines or vertical segments from the null vertical slabs and that are not contained in any null vertical slab. We call *opposite cells* two cells (an upper cell and a lower cell) that have intersection with some common vertical line. Note that two blue segments of opposite cells that have intersections with some common vertical line, (*opposite blue segments*), determine in the primal plane two blue points. This blue points have the property that there exist two parallel lines passing through the blue points and defining a strip interval, since in the dual plane *between the blue segments* there are only red lines and no blue lines. The maximum width of the strips in this interval can be determined in constant time. To compute the widest strip we make a plane-sweep with a vertical line halting at the opposite blue segments. Each stop we compute in constant time the maximum width of the strip interval and update the maximum.

Let $A_U(n,m)$ ($A_L(n,m)$) be the ordered set of segments that define upper cells (lower cells) generated by the arrangement of n blue lines and m red lines. Let $c_U(n,m)$ ($c_L(n,m)$) be the respective cellular complexity, i.e., the number of segments in all the upper cells (lower cells). We define $A(n,m) = A_U(n,m) \cup A_L(n,m)$ and $c(n,m) = c_U(n,m) + c_L(n,m)$. We will show that $c(n,m)$ is not large and that we can generate $A(n,m)$ in a suitable time.

Lemma 3. *The number of upper and lower cells generated by the arrangement of m red lines and n blue lines is less or equal to $2n+2$.*

Proof. We count only the number of upper cells. Analogously we proceed with the lower cells. Initially, the region U is a single cell. The first blue line divide the initial cell in two cells. Therefore, if d_n represent the number of upper cells after the introduction of n blue lines, we have that $d_1 = 2$. Suppose we have introduced $n-1$ blue lines by increasing slope beginning from the vertical. The *null vertical slab* of the following blue line can divide at most a upper cell in two cells. In any other case the blue line only modify or delete upper cells. Therefore, $d_n \leq d_{n-1} + 1$, and then $d_n \leq n+1$. □

Lemma 4. *The cellular complexity is $c(n,m) = O(N)$.*

Proof. In the cellular complexity we consider all the segments that define upper and lower cells: vertical segments, red segments and blue segments. We count separately each kind of segments and only for the upper cells. For the lower cells we proceed analogously.

i) The number of vertical segments is less or equal than n, since they are generated by the blue lines and each blue line can produce at most one vertical segment in U (null vertical slab).

ii) Initially, there are at most m red segments. A blue line can divide a red segment in two segments, increasing the number in one. Therefore the number of red segments is less or equal to $m+n$.

iii) Now, we count the blue segments. In the arrangement of the upper cells, a blue line contributes with at most one segment or with two or more segments. Note that in one upper cell there are not two segments from the same line, and also that two lines *share* segments in at most one cell. By the above lemma, the number of upper cells is less or equal to $n+1$. The lines that contribute with only one segment, give an over all contribution of at most n segments. The lines that contribute with two or more segments each are counted in the following way:

1) Lines with slope between a_{q_1} and the vertical, ordered by decreasing slope in this interval: let $k_1, k_2, \ldots, k_s$ be the number of segments contributed by each line. It is clear that a line can share a segment with one previous line in at most one cell, since the segment contribution is from the last intersection with the previous lines and therefore, only with the last previous line can share a segment in a cell. If we count cells, in the worse case we'll have that $k_1 + k_2 - 1 + \ldots + k_s - 1 \leq n+1$, and then $k_1 + k_2 + \ldots + k_s \leq n + s$ (see Figure 18).

2) Lines with slope between the vertical and a_{q_m}, ordered by increasing slope in this interval. Let $k'_1, k'_2, \ldots, k'_t$ be the number of segments contributed by each line. With a similar reasoning we obtain $k'_1 + k'_2 + \ldots + k'_t \leq n + t$. Taking into account that $s + t \leq n$, we can deduce that the total number of segments in the upper cells is less or equal to $6n + m$. □

Lemma 5. *$A(n,m)$ can be computed in $O(N \log n)$ time.*

Proof. $A_U(n,m)$ is computed by divide and conquer in time $T(n) = 2T(n/2) + f(n,m)$, where $f(n,m) = O(N)$ is the time needed to merge the cells of the two subproblems, since the number of segments of the cells of the subproblems is $O(n+m)$, making a plane sweep we get the resulting cells. Analogously for $A_L(n,m)$. □

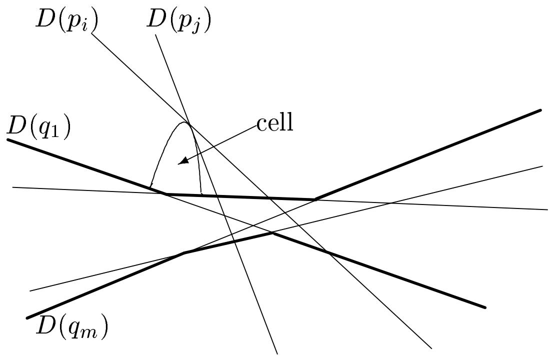


Figure 18: Segments of a cell.

Theorem 6. *Let P and Q be two disjoint strip separable sets of n and m points in the plane respectively. The widest strip (or strips) can be found in $O(N \log N)$ time.*

Proof. By the above discussion and using the previous lemmas, we can obtain $A(n,m)$ in $O(N \log n)$ time. We make a plane sweep with a vertical line halting in the $O(n)$ opposite blue segments [9]. □

3.2 Separating segments by strips

This problem can be solved analogously to the strip separability problem for points. The strip intervals can be obtained from the final polygonals constructed for the wedge separability of segments. Each polygonal of the final polygonals defines an angular interval in $[0, 2\pi)$ where there are not slopes of strips and it is called the *negative interval*. Using the final polygonals and in $O(n\alpha(n))$ time we obtain the at most n disjoint negative intervals in $[0, 2\pi)$. We compute their union and reduce it to the interval $[0, \pi)$. The strip intervals are the complement of the union with respect to the interval $[0, \pi)$. There are at most n strip intervals and they are closed intervals or points.

Proposition 6. *Let P and Q be two disjoint sets of n and m segments in the plane respectively. To decide whether they are strip separable computing the strip intervals can be done in $O(N \log N)$ time.*

3.2.1 The narrowest strip

The narrowest strip depends on $CH(Q)$ and on the strip intervals. We have to compute the width of the convex polygon $CH(Q)$, i.e., the small distance between support parallel lines [14,17] with the restriction to the strip intervals. It is similar to the case of the point sets.

Proposition 7. *Let P and Q be two disjoint strip separable sets of n and m segments in the plane respectively. If we have pre-computed the strip intervals, the narrowest strip can be found in $O(N)$ time.*

3.2.2 The widest strip

The widest strip depends on the segments of P and on the strip intervals. We proceed analogously as in the case of points sets. If all the strip intervals are points, in time $O(n)$ we analyze each one in constant time and determine the widest strip. Otherwise we choose a strip interval $[\alpha_1, \alpha_2]$, $\alpha_1 \neq \alpha_2$ and we make a change of coordinates taking as vertical a line with slope within (α_1, α_2) and as origin of coordinates a point such that the segment extremes of $P \cup Q$ have different abscissa (see Figure 19). We consider the problem in the dual plane by doing the transformation:

$$L : y = 2ax - b \longleftrightarrow p : (a, b), \qquad D(L) = p, D(p) = L.$$

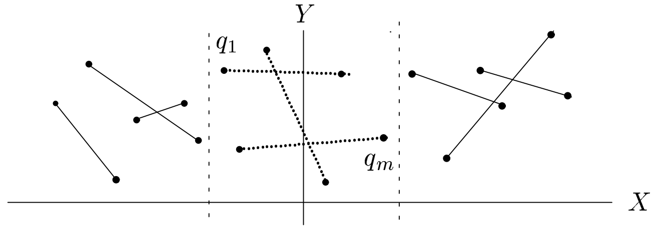


Figure 19: Changing coordinates.

We transform the points of $CH(Q)$ and the extremes of segments of P having red and blue lines. The blue lines in the dual plane have different slopes within the slope interval (a_{q_1}, a_{q_m}), where q_1 and q_m are the red points of minimum and maximum abscissa respectively.

With the red lines we compute the upper envelope, E_U, and the lower envelope, E_L, in $O(m \log m)$ time. The dual of a segment of P located on the left (right) side of the axis Y gives in the dual plane a double wedge defined by the two lines with negative (positive) slope that are the dual of the segment extremes. The dual arrangement is similar to the case of points with the restriction of the double wedge regions that can be controlled by *null vertical slabs* defined by the intersection points of the two lines of the double wedge with E_U and E_L (see Figure 20).

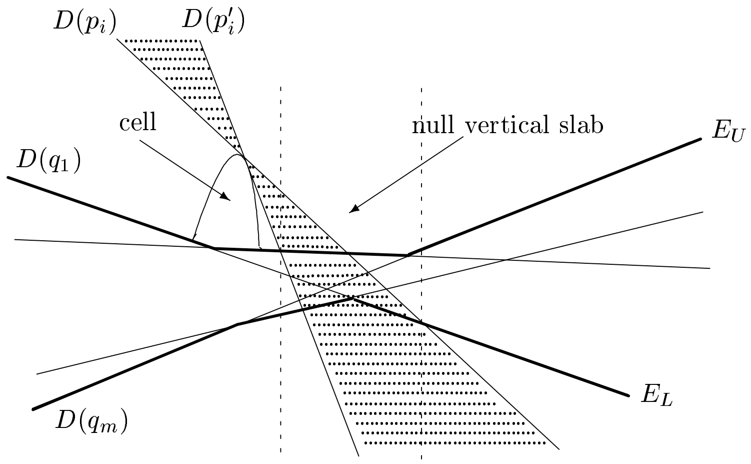


Figure 20: Dual of segments.

Therefore, the number of upper (lower) cells and the cellular complexity is less or equal than the corresponding values obtained in the case of the widest strip for points. It is clear that we can apply a similar algorithm to determine the widest strip in $O(N \log N)$ time.

Theorem 7. *Let P and Q be two disjoint strip separable sets of n and m segments in the plane respectively. The widest separating strip can be found in $O(N \log N)$ time.*

3.3 Separating polygons by strips

The problem of the strip separability for polygons is easily reduced to the problem of the strip separability for segments described in 3.2. Hence we have:

Corollary 2. *Let P and Q be two disjoint sets of polygons in the plane with n and m segments respectively in total. The strip intervals and the narrowest and the widest strip can be found in $O(N \log N)$ time.*

3.4 Separating circles by strips

Using the final polygonals constructed in the wedge separability of circles, we compute the union of the negative intervals of the polygonals of the final polygonals. The complement of this union with respect to $[0, \pi)$ is the set of at most n closed strip intervals or points.

Proposition 8. *Let P and Q be two disjoints set of n and m circles of the plane respectively. To decide whether they are strip separable computing the strip intervals can be done in $O(N \log N)$ time.*

3.4.1 The narrowest strip

The narrowest strip depends on $CH(Q)$ and on the strip intervals. To compute the strip with the minimum width we use calipers polygon [14,17]. Starting on a segment of $CH(Q)$ we move the caliper (two parallel support lines) in counter clock-wise with the minimum angle taking into account the restriction of the strip intervals and determining antipodal pairs of circular arcs (see Figure 21 a)). This process can be done in $O(m + n)$ time [13]. For a pair of antipodal circular arcs the minimum width (distance between the support lines) corresponds to the extremes of the circular arcs since their chords are parallel and the variation of the width increases and decreases (see Figure 21 b)). Analyzing in constant time all the antipodal pairs of circular arcs we computation the narrowest strip in $O(N)$ time.

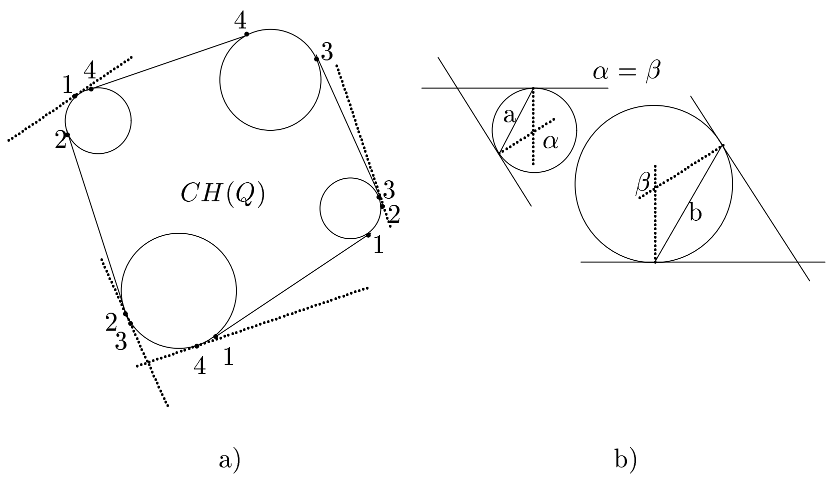


Figure 21: a) Computing antipodal arcs, b) antipodal arcs.

Proposition 9. *Let P and Q be two disjoint strip separable sets of n and m circles in the plane respectively. If we have pre-computed the strip intervals, the narrowest strip can be found in $O(N)$ time.*

3.4.2 The widest strip

The widest strip depends on the circular arcs of the final polygonals. If all the strip intervals are points, in time $O(n)$ we analyze each one in constant time and determine the widest strip. Otherwise we choose a strip interval $[\alpha_1, \alpha_2]$, $\alpha_1 \neq \alpha_2$ and we make a change of coordinates taking as vertical a line with slope within (α_1, α_2) and as origin a point such that the extremes of the blue circular arcs of the final polygonals and the extremes of the red circular arcs of $CH(Q)$ have different abscissa (see Figure 22 a)). We consider a correlation of the projective plane in his dual such that the tangents to a circle are transformed into the points of a hyperbole. Any vertical line in the dual plane intersects the hyperbole in two points, so we can talk about the upper branch and the lower branch of the hyperbole. The dual of the points of a circle are the tangent lines to a hyperbole. The dual of the points of a circular arc correspond to tangent lines to a hyperbolic arc (see Figure 22 b)). We proceed in the following way:

Dual of $CH(Q)$: each of the m red circles gives in the dual plane two hyperbolic branches. Each branch is an unbounded x-monotone Jordan curve and any pair of them intersects at most in two points. We compute the upper (lower) envelope E_U (E_L) of the at most $2m$ hyperbolic branches using the theorem 6.1 of [15] in $O(\lambda_2(m) \log m) = O(m \log m)$ time. The complexity of E_U and E_L is less than or equal to the complexity of a single face of $2m$ unbounded Jordan curves which is $O(\lambda_2(m)) = O(m)$ (see theorem 5.7 of [15]).

Dual of blue circles: first, we transform to dual plane the circles of P that contribute with at least one circular arc to the final polygonals. In the dual plane we compute the intersections of each blue hyperbolic branch with E_U and with E_L, obtaining the null vertical slabs formed by the vertical parallel lines that passed through the intersection points. We compute the union of the intervals defined by the null vertical slabs in order to get the disjoint set of the null vertical slabs. In this slabs there are not upper or lower cells. All these computations can be done in $O(n \log N)$ time.

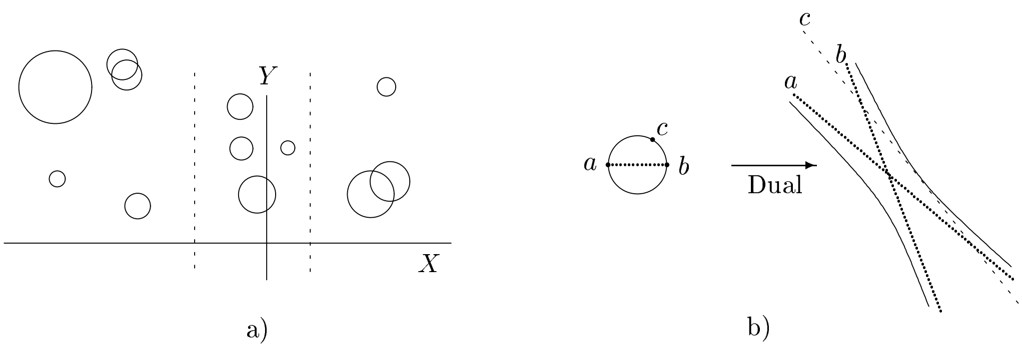


Figure 22: a) Change of coordinates, b) dual of a circle.

Next, we dualize only the circular arcs of the final polygonals obtaining tangent lines to hyperbolic arcs. From the hyperbolic arcs we take only the parts of the arcs that are over E_U, arrangement HA_U, and the parts of the arcs that are under E_L, arrangement HA_L. The complexity of the hyperbolic arcs is $O(n)$, since the complexity of the final polygonals of circular arcs is also $O(n)$.

The arrangement of the upper (lower) cells can be obtained computing the lower (upper) envelope of HA_U (HA_L) restricted to the union of null vertical slabs. The intersection point of two hyperbolic arcs in the dual plane corresponds in the primal plane to a common tangent line to the circular arcs.

Given a circle R and an circle C exterior to R, we say that a point p of C *see* R if the region defined by the half line tangents from p to R and the part of R between the half line tangents does not contain any obstacle. A (connected) circular arc a of C *see* R if all the points of a see R.

Lemma 6. *Let C_1 and C_2 be two blue circles exterior to a red circle R and let a_1 and a_2 be circular arcs of C_1 and C_2 respectively that see R. At most one of tangents between C_1 and C_2 passed through points of a_1 and a_2 simultaneously.*

Proof. The arc a_1 (a_2) is contained in the arc obtained with the interior tangents between C_1 (C_2) and R (see Figure 23 a)). Any tangent line to C_1 (C_2) at a point of the arc a_1 (a_2) separates C_1 (C_2) from R.

The interior tangents of C_1 and C_2 do not passed through points of a_1 and a_2: if C_1 does not intersects C_2 and p_1 and p_2 are the tangency points of an interior tangent then p_1 and p_2 are not visible from R

simultaneously since their *visibility regions* are disjoint (see Figure 23 a)). If C_1 intersects C_2, there are not interior tangents. If C_1 is tangent to C_2, the tangency point has null visibility.

At most one exterior tangent of C_1 and C_2 passed through points of a_1 and a_2 simultaneously. Let p_1, p_2 be the tangency points of one exterior tangent between C_1 and C_2, and q_1, q_2 the tangency points of the other exterior tangent between C_1 and C_2: if the exterior tangents are parallel, the visibility regions of p_1 and p_2 and of q_1 and q_2 are disjoint. If the exterior tangents intersect, the visibility region of p_1, p_2, q_1 and q_2 is the wedge formed by the exterior tangents. If R is within this region, there are three different situations:

1) The exterior tangents are tangent to R: one of the blue circles is not visible from R.

2) One exterior tangent is tangent to R and the other one is not: a_1 and a_2 share only the another exterior tangent.

3) The exterior tangents are not tangent to R: The farthest blue circle from R has two disjoint visibility arcs, each one share an exterior tangent with the visibility arc of the nearest circle to R (Figure 23 b)). □

The lemma is also true for convex closed curves each pair of which intersects in at most two points. From the above lemma, we ensure that each pair of hyperbolic arcs intersects in at most one point and therefore, we can compute the lower envelope of HA_U (upper cells) and the upper envelope of HA_L (lower cells) in $O(\lambda_{1+1}(n) \log n) = O(n \log n)$ time.

The complexity of the upper and lower envelopes is $O(n\alpha(n))$. Making a plane sweep with a vertical line on the cells we determine the at most $O(n\alpha(n))$ pairs of circular arcs that define strips. In constant time we compute the maximum width of the interval strip of each pair of circular arcs. The maximum width corresponds to the intersection points of the circles with the line that passed through both centers of the circles in the case that this intersection points belongs to the pair of circular arcs or it corresponds to one common extreme of the pair of circular arcs. Therefore, in $O(n\alpha(n))$ time we compute the widest strip.

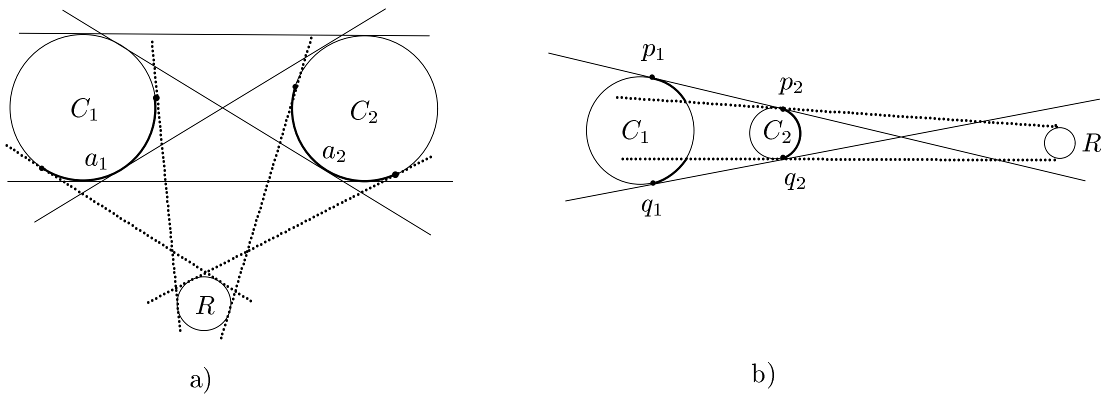


Figure 23: Tangents between two circles.

Theorem 8. *Let P and Q be two disjoint strip separable sets of n and m circles in the plane respectively. The widest strip can be found in $O(N \log N)$ time.*

4 Conclusions

We have solved the problems of separating two sets of points, segments, polygons and circles in the plane with wedges and strips, obtaining also the wedges with the maximum and minimum angle and the narrowest and the widest strip, our algorithms running in $O(N \log N)$ time.

In case that there no separating wedges or strips we are currently studying the separability problems from other points of view, either by weakening the separability condition, allowing some points to be missclassified, or by using more than one separator. We are also extending our results to $3D$.

References

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

[2] J.-D. Boissonnat, J. Czyzowicz, O. Devillers, J. Urrutia, M. Yvinec, *Computing Largest Circles Separating Two Sets of Segments*, Proceedings of the Eighth Canadian Conference on Computational Geometry, 1996, pp. 173-178.

[3] 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.

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

[5] S. Fish, *Separating Point Sets by Circles and the Recognition of Digital Disks*, Pattern Analysis and Machine Intelligence 8 (4), 1986.

[6] J. Hershberger, *Finding the upper envelop of n line segments in $O(n \log n)$ time*, Information Processing Letters, 33, 1989, pp. 169-174.

[7] M. E. Houle, *Weak separation of sets*, Ph. D. Thesis, McGill University, 1989.

[8] F. Hurtado, M. Noy, C. Seara, *Dominación Geométrica respecto de un Convexo*, manuscrito.

[9] F. Hurtado, M. Noy, C. Seara, *Separación de puntos mediante cuñas y bandas*, VII Encuentros de Geometría computational, Madrid, 1997, pp. 45-55.

[10] N. Megiddo, *Linear-time algorithms for linear programming in $\mathbb{R}^3$ and related problems*, SIAM J. Computer, Vol 12, No 4, 1983, pp. 759-776.

[11] J. O'Rourke, S. R. Kosaraju, N. Megido, *Computing Circular Separability*, Discrete Comput. Geom. 1 (1), 1986, pp. 105-113.

[12] D. Rotem, J. Urrutia, *Circular Permutations Graphs*, Networks, 11, 1982, pp. 429-437.

[13] D. Rappaport, *A convex hull algorithm for discs, and applications*, Computational Geometry: Theory and Applications, 1, 1992, pp. 171-187.

[14] M. I. Shamos, *Computational Geometry*, Ph. D. Thesis, Yale University, 1978.

[15] M. Sharir, P. K. Agarwal, *Davenport-Schinzel sequences and their geometric applications*, Cambridge University Press. 1995.

[16] J. Stoer, C. Witzgall, *Convexity and Optimization in Finite Dimension I*, Springer-Verlag, New York, 1970.

[17] G. T. Toussaint, *Solving Geometric Problems with Calipers*, Proceedings of IEE MELECON'83, 1983.