Separability by two lines and by flat polygonals*

Ferran Hurtado† Mercè Mora† Pedro A. Ramos‡ Carlos Seara†

25th June 2002

Abstract

In this paper we study the separability in the plane by two criteria: *double wedge separability* and *constant turn separability*. We give $O(N \log N)$-time optimal algorithms for computing all the separating double wedges of two disjoint sets of objects (points, segments, polygons and circles) and the flattest minimal constant turn polygonal line separating two sets of points.

1 Introduction

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

Let $\mathcal{C}$ be a family of curves in the plane. The sets B and R are $\mathcal{C}$ *separable* if there exists a curve $S \in \mathcal{C}$ such that every connected component of $\mathbb{R}^2 - S$ contains objects only from B or from R. If S is a straight line, the sets B and R are *line separable.* The decision problem of linear separability for any of the above object classes can be solved in $O(N)$ time [10, 11]. The region of the plane formed by the points of the separating lines can be computed in $\Theta(N \log N)$ time.

Many alternatives have been considered when linear separability is not possible like using polygonal lines or other geometrical objects [4, 7, 9, 10]. In [9] it is considered the *wedge separability* and the *strip separability.* In the first case, S is a wedge and in the second one S is a strip (the region between two parallel lines). The region of the plane formed by the vertices of separating wedges and the wedges with maximum and minimum angle are computed in $O(N \log N)$ time. If the angle of the wedge is exactly π, the problem reduces to linear separability. The intervals of the slopes of all the separating strips, and the narrowest and the widest strip are computed in $O(N \log N)$ time.

If S is a closed curve there are also different kinds of separability. The problem of finding the minimum (in the number of edges) convex polygon separating two point sets in the plane is considered in [7], and the problem is solved in $O(N \log N)$ time. If the objects are segments, polygons or circles, the problem is solved in [9] with the same complexity bound. Finally, [4] deals with the problem of computing all the largest circles separating two sets of segments.

In case that B and R are not wedge separable, we want to know what is the minimum number of wedges separating B and R. A simpler problem is knowing whether the minimum number is two and a particular case is the double wedge separability (Figure **??**a).

*Partially supported by Proyectos DGES-MEC PB98-0933 and Gen. Cat. 1999SGR000356.

†Dpt. Matemàtica Aplicada II, Universitat Politècnica de Catalunya, Pau Gargallo 5, 08028 Barcelona, Spain, e-mail: seara@ma2.upc.es

‡Dpto. de Matemáticas, Universidad de Alcalá, Aptdo. de Correos 20, 28871 Alcalá de Henares, Madrid, Spain

Definition 1. *Two disjoint sets of objects B and R are double wedge separable if there exist two straight lines intersecting at a point p, vertex of the double wedge, such that $\{B_1, B_2\}$ is a partition of B, $\{R_1, R_2\}$ is a partition of R and the subsets alternate angularly as seen from p.*

Another criteria of separability is the *radial separability*: given B, R and a point p, we order radially the points of $B \cup R$ obtaining a separation of $B \cup R$ by sectors. The corresponding optimization problem would be computing the point p from where we see radially $B \cup R$ with the minimum number of sectors. This problem generalizes some of the problems mentioned earlier because if the minimum is two, the sets are wedge separable, while if the minimum is four and the sectors are defined by two intersecting lines, then the sets are double wedge separable.

In Section 2 we solve the problem of the double wedge separability of two disjoint sets of objects in the plane given an $O(N \log N)$ time algorithm for computing the region formed by the vertices of the double wedges. The algorithm is optimal because the $\Omega(N \log N)$ lower bound for the decision problem is shown in [1]. The double wedge separability is also meaningful in case that the sets are wedge separable because the regions of vertices of wedges and double wedges are disjoint by definition. The key idea for the algorithm is to observe that if the points are double wedge separable, there exists a vertical line or a horizontal line that gives the same partition of one of the sets as the double wedge. Using this idea, a number of candidate partitions are generated and for each partition the (possibly empty) region of vertices of the corresponding double wedges is computed. The locus of vertices of the double wedges is formed by a linear number of quadrilaterals. Finally, we study the double wedge separability for different kinds of objects.

If the sets B and R are not line separable a natural question is to ask about the separability by a polygonal line. In [8] the author studied the *minimum-link red-blue separation* problem (find a polygonal separator with fewest edges that separates B from R). This problem is known to be NP-complete. In particular, we are interested in the separability by polygonal lines of constant turn, which we call *constant turn separability* (Figure **??**b).

Definition 2. *A Θ-polygonal is a polygonal line such that all the corners have angle Θ and it turns alternatively left and right. Two disjoint sets of objects B and R are constant turn separable if there exists a Θ-polygonal separating B and R.*

In Section 3 we study the constant turn separability of two disjoint sets of points in the plane. We give $O(N \log N)$ time algorithms for computing the separating Θ-polygonals with maximum angle and also with the minimum number of edges. We observe that two given sets B and R can always be separated by a Θ-polygonal for Θ small enough. However, we are interested in maximizing the angle because if Θ is close to π we get an approximation of linear separability.

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 quadrant, because this situation is a special case of wedge separability which can be detected directly with the algorithms in [9]. Given a double wedge ω with vertex at p and separating B and R, the plane is decomposed into two complementary cones with apex p, the *red cone* and the *blue cone*, 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) cone (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.

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 Ω; the total set 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 semicones 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 semicone 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 semicone partitions for B and R, we consider the monotone polygonal lines $\mathcal{P}_B$, with vertices $\{b_1, \ldots, b_n\}$ and $\mathcal{P}_R$, with vertices $\{r_1, \ldots, r_m\}$. Each edge of a polygonal 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 $(\mathcal{P}_R \setminus e_r) \cap (\mathcal{P}_B \setminus e_b) = \varnothing$. Furthermore, at least one of the following things 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 lines is odd (Figure **??**). 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. ☐

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, 5] 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., $R_1 = \{r_1, \dots, r_j\}$ and $R_2 = \{r_{j+1}, \dots, r_m\}$. For $i = 1, \dots, n-1$, let $B_i = \{b_1, \dots, b_i\}$ and $B'_i = \{b_{i+1}, \dots, b_n\}$ 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 precomputed.

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 B_{i+k} is neither 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 **??**).

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 **??**). 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 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].

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 [13].

2.1 Double wedge with maximum aperture angle

By *aperture angle* of a double wedge we mean $\max\{\alpha_r, \alpha_b\}$. Observe that as $\alpha_r + \alpha_b = \pi$ maximizing and minimizing the aperture are the same problem. The region of vertices of the separating double wedges is the union of a linear number of (possibly degenerate) quadrilaterals. In each quadrilateral, the vertex of the double wedge with maximum aperture must be one of the four vertices, as otherwise we could move the vertex of the double wedge along the boundary of the quadrilateral increasing one of the angles and decreasing the other. Once all the quadrilaterals have been constructed, we can compute the angles in the vertices of all them and hence obtain the separating double wedge with maximum aperture in additional linear time. Therefore, we have:

Proposition 1. *Given the locus of vertices of double wedges separating B and R, the double wedge with maximum aperture angle can be computed in $O(N)$ 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 quadrant, say, might be crossed by a red segment with endpoints on the red wedges. We again look for a separating double wedge where the vertical direction lies in the blue semicone, 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 polygonals joining endpoints as we did for sets of points. Only edges of the red polygonal line, 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 this edges *critical red bridges* (Figure **??**). *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 wedge separable can be done in $O(N \log N)$ time. The locus of separating double wedge vertices can be constructed within the same time bound, and the separating double wedge with maximum aperture angle can be computed in $O(N)$ additional time once these regions are available.*

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 **??**). We are showing 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 [6]. Therefore we obtain:

Theorem 3. *Let C_B and C_R be disjoint sets of circles in the plane. Whether they are double wedge separable can be done in $O(N \log N)$ time. The locus of separating double wedge vertices can be constructed within the same time bound, and the separating double wedge with maximum aperture angle can be computed in $O(N)$ additional time once these regions are available.*

3 Constant turn separability

When the sets B and R are not linearly separable we may be interested in finding a polygonal line separating the sets and satisfying some restrictions. The problem of the polygonal separator with the minimum number of edges is NP-complete [8]. A simpler problem is the *constant turn separability*, i.e., the separability by a polygonal line that turns alternatively left and right with angle Θ, *a Θ-polygonal line.* In particular, we would like to maximize the angle Θ because the closer to π the better the approximation to linear separability.

It is easy to see that two point sets B and R are always separable by some Θ-polygonal line 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 line separating B and R because this a necessary step for our later addressing the problem of computing a Θ-polygonal line separator with maximum angle.

Computing a Θ_0-polygonal line separator

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

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 **??**).

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 line 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 constant turn separable by some Θ_0-polygonal line, and one such separator can be constructed in $O(N \log N)$ time.*

Computing a separating Θ-polygonal line 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 plane 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* **??***).*

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* **??***).*

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 $O(\frac{n}{\Theta} \log n)$ 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.

Figura de los maximales coloreados

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

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* **??***).*

With this definition it is clear that once we have computed a Θ_0-polygonal line 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 plane 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.

- 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 which are apices of wedges which 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. For $\beta \geq \frac{\pi}{2}$ the ray $R_{p,e}$ with origin p perpendicular to e must be between L_p and R_p; the procedure sorts the candidate points by their orthogonal projection onto e, and then two sweeps by lines perpendicular to e allow an incremental convex hull computation which gives the rays L_p and R_p as side product(Figure **??**). 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.

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 line 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 which 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 line.

Procedure Θ-Polygonal-Line
Input: B, R, Θ_0,
Output: A separating Θ-polygonal line with maximum $\Theta = \Theta_M$.

1. Compute $CH(B)$, let $\{e_1, \ldots, e_k\}$ be the sorted list of edges of $CH(B)$. Classify the red points into: *interior* to $CH(B)$, R_I, and *exterior or border* to $CH(B)$, R_E.

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 $B \cup R_I$ which are candidates for each edge of $CH(B)$. A point can be 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 $\frac{2\pi}{\Theta_0}$.

   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 which are unoriented β-maxima with respect to B for some $\beta \geq \Theta_0$ and all wedges which 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 which support $CH(B)$, because the external angles β_i they define (always greater or equal than π) are also making these points unoriented maxima (see Figure XXX).

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. If we place these values on

concentric circles around a point (Figure **??**), the intersection of the 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 line for d_1 in $O(N \log N)$ time analogously as we did in Proposition 2. (Polygonal lines 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 running time $O(N \log N)$ 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 line with maximum $\Theta = \Theta_M$ can be achieved in $O(N \log N)$ time.*

Maximum angle and minimum number of corners

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

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 line with "valley" vertices on red points (Figure **??**) obtained by applying the construction from Proposition 2. Let $\mathcal{P}_{\Theta_M,B}$ be the separating Θ_M-polygonal line with "top" vertices on blue points (Figure **??**) 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.

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

Proof. Let $\mathcal{P}_M$ be a separating Θ_M-polygonal with minimum number of corners for the direction d_1, which will consist of two half lines and a set of edges. Assume without loss of generality 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 can disappears during this process because of the assumption of initial minimality. At the end, the resulting polygonal fulfills the conditions stated in the lemma. □

The above Lemma provides the following greedy approach to find a separating Θ_M-polygonal with minimum number of corners 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 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 halflines, and exit with the solution giving smallest number of corners among these four possibilities.

We know that $\mathcal{P}_{\Theta_M,B}$ and $\mathcal{P}_{\Theta_M,R}$ can be constructed in $O(N \log N)$ 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 (one per each bisector d_i), we can state the following theorem:

Theorem 6. *Given two disjoint points sets B and R in the plane the max-angle min-corner polygonal line separating the sets can be computed in $O(N \log N)$ time.*

Lower bound

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

Theorem 7. *The problem of computing the Θ-polygonal line with maximum $\Theta = \Theta_M$ separating two disjoint points 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 [14].

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 $i = 1, \ldots, N$.

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 **??**). Hence, the sets R and B are not separable by any Θ-polygonal line with $\pi/2 \leq \Theta < \pi$ because in that situation all red points would be Θ-unoriented maxima with respect to B.

Reversely, 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 line 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 line with $\pi/2 \leq \Theta < \pi$, and the claimed reduction is proved. □

Acknowledgement. We are very grateful to Prof. Joseph Mitchell for helpful discussions on earlier versions of this paper.

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. J. Comput. 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, **28**, 1979, pp. 643–647.

[4] 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.

[5] B. M. Chazelle, H. Edelsbrunner, *An Optimal Algorithm for Intersecting Line Segments*, Proc. 29th IEEE Symp. on Fund. of Computer Science, Oct. 1988.

[6] 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.

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

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

[9] 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.

[10] M. E. Houle, *Weak Separation of Sets*, Ph. D. Thesis, McGill University, 1989.

[11] 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.

[13] C. Seara, *On Geometric Separability*, Ph. D. Thesis, Univ. Politècnica de Catalunya, in preparation.

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