

Red-Blue Separability Problems in 3D ^{*}

Ferran Hurtado [†] Carlos Seara [†] Saurabh Sethia [‡]

1 Introduction

Let B and R be two disjoint sets of points in 3D classified as *blue* and *red* points, respectively. Let n be the number of points in B and R . We consider the sets of points in general position, thus there are no four points in a plane and no three points on a line. Let \mathcal{C} be a family of surfaces in 3D. The sets B and R are \mathcal{C} *separable* if there exists a surface $S \in \mathcal{C}$ such that every connected component of $\mathbb{R}^3 - S$ contains points only from B or from R . If S is a plane, we have *linear separability*. The decision problem of linear separability for two disjoint sets of objects (points, segments, polygons or circles) in 2D or (points, segments, polyhedra or spheres) in 3D can be solved in linear time [8, 13].

In [1, 10, 11] the authors study the separability of two disjoint point sets in the plane by the following criteria: *wedge separability*, *strip separability* and *double wedge separability*. Optimal $\Theta(n \log n)$ time algorithms for deciding wedge, strip, and double wedge separability, as well as for constructing the locus of feasible apices of wedges and double wedges, and the interval of feasible slopes of strips are described in [1, 10, 11]. They have also shown how to find wedges and double wedges with maximum and minimum aperture angle, and the narrowest and the widest strips.

In this abstract we summarize results from our work on separability of two disjoint point sets in 3D that extends the criteria above and gives solutions to various separability problems. For each separability criterion we consider the problem of deciding whether that particular separability is feasible, which is probably equivalent to finding one solution to the problem; in some cases, we also consider the problem of finding partial descriptions of all feasible solu-

tions. For some separability criteria we consider, the convex hull of either the red or blue points needs to be monochromatic. If it is so, we perform this check as a first step in our algorithms in $O(n \log n)$ time by computing $CH(R)$ and $CH(B)$ [14]. First, we consider the problem of computing all the feasible solutions for linear separability. Second, we study slice, wedge, and diwedge separability as the natural extensions in 3D for strip, wedge, and double wedge separability in 2D. Third, we study the decision problem for prismatic, pyramidal, and dipyramidal separability, which also can be considered as extensions in 3D for strip, wedge, and double wedge separability criteria, but allowing a linear number of planes. Finally, we study some separability criteria defined by a constant number of planes.

We provide proof for one of our results (slice separability) and state others without proof due to space constraint.

2 Linear separability

The decision problem of linear separability for two disjoint point sets in 3D can be solved in linear time [13]. The problem of computing all feasible solutions is solved by the next theorem.

Theorem 1. *The locus of all the planes separating B and R can be computed in $\Theta(n \log n)$ time. Once we have pre-computed the locus, to decide whether a given plane separates the point sets can be done in $O(\log n)$ time.*

The problem of computing the maximum Euclidean distance between two parallel separating planes has been solved by Houle [8] with the following theorem.

Theorem 2. [8] *Given two point sets in \mathbb{R}^d , then a separating hyperplane which minimizes*

^{*}Partially supported by grants DGES-MEC-SEUID PB98-0933, MCYT-FEDER BFM2002-0557, DURSI 2001SGR00224 and Joint Commission USA-Spain for Scientific and Technological Cooperation Project 98191.

[†]Dpt. Matemàtica Aplicada II, U.P.C., Barcelona, Spain, {hurtado, seara}@ma2.upc.es

[‡]Department of Computer Science, Oregon State University, Corvallis, OR 97331, USA, saurabh@cs.orst.edu

the orthogonal Euclidean distance between the hyperplane and the point sets may be found, or its non-existence determined, in $O(n)$ time.

A consequence is the following corollary.

Corollary 1. *The widest separating slice defined by two parallel separating planes of B and R can be computed in $O(n)$ time.*

3 Separability by two planes

In this section we study slice, wedge, and di-wedge separabilities, which involve exactly two separating planes.

3.1 Slice separability

A slice is defined as the space between two parallel planes. The normal vector to the planes of the slice is called the *slice direction*. The slice separability problem asks the question: Is there a slice that contains all the red points but does not contain any blue point or vice versa?

Theorem 3. *Deciding whether the sets B and R are slice separable can be done in $O(n^3)$ time. The locus of slice directions of all feasible solutions can be computed in $O(n^3 \log n)$ time and the complexity of the locus is $\Theta(n^2)$.*

Proof. (Sketch) Let u be the *slice direction* of a separating slice which contains R . Let B_1 (B_2) be the set of blue points above (below) the slice. A plane π_1 with normal vector u passing through any red point r is a separating plane of B_1 and B_2 . While keeping the point r on π_1 , we can move u around with two degrees of freedom until it bumps into two blue points b_1 and b_2 . Thus, a separating slice (if it exists) can be found by the following $O(n^3)$ time algorithm.

1. Choose a red point $r \in R$.
2. For each pair of blue points b_1 and b_2 , compute the plane π_1 passing through b_1 , b_2 , and r . Compute B_1 (B_2), the set of blue points above (below) π_1 .
3. By linear programming compute a separating slice (if it exists) defined by two parallel planes such that one separates B_1 from $R \cup B_2$ and, the other one separates $R \cup B_1$ from B_2 .

In order to compute the set of slice directions of all the feasible solutions we proceed as follows.

- 3.1. Compute $CH(B_1)$, $CH(B_2)$, $CH(R \cup B_1)$, and $CH(R \cup B_2)$.

- 3.2. Compute the region on the unit sphere formed by the set of directions of the separating planes between $CH(B_1)$ and $CH(R \cup B_2)$. Proceed analogously with $CH(R \cup B_1)$ and $CH(B_2)$. We obtain two regions each bounded by at most two convex chains with $O(n)$ complexity. The projection of each region on the plane $z = 1$ is either a convex polygon or two unbounded convex polygons.

- 3.3. Compute the intersection of the two regions in $O(n)$ time. Its boundary corresponds to parallel planes which are always touching $CH(R)$ and some blue point.

Thus, in additional $O(n \log n)$ time, we can compute the convex region on the unit sphere defined by the set of slice directions of all feasible solutions for each good partition. In full paper, we also show that the locus on the unit sphere of the set of slice directions of all feasible solutions is formed by at most $O(n^2)$ connected components, each one with at most linear complexity, and the total complexity of the locus is $\Theta(n^2)$. \square

The width of a separating slice is defined by the distance between the two parallel planes. It is natural to ask about the narrowest and the widest separating slices.

Theorem 4. *Once we have pre-computed the set of slice directions of the separating slices for B and R , the widest and the narrowest separating slices can be computed in $O(n^3)$ and $O(n^2 \log n)$ time, respectively.*

3.2 Wedge separability

Two intersecting planes divide the 3D space into four quadrants. Any one quadrant is called a 3D wedge. The wedge separability problem is the following: Is there a wedge that contains R but does not contain any blue point or vice versa?

Theorem 5. *Deciding whether B and R are wedge separable can be done in $O(n^3)$ time.*

For wedge separable point sets we ask about a separating wedge with maximum or minimum aperture angle. The wedge with minimum aperture angle is related to slice separability (a slice can be considered as a wedge with aperture angle 0°) and the wedge with maximum aperture angle is related to linear separability (if the aperture angle is close to 180° then, we have a good approximation to linear separability).

Theorem 6. *Computing a separating wedge for B and R with maximum or minimum aperture angle can be done in $O(n^3 \log n)$ time.*

A natural problem is to decide whether there exists a separating wedge with a fixed aperture angle θ , $0^\circ \leq \theta \leq 180^\circ$. Note that if we have pre-computed the separating wedges with maximum and minimum aperture angle, the problem is not solved because it may be that B and R are not wedge separable for all possible values of aperture angle between the minimum and the maximum.

Theorem 7. *Computing a separating wedge for B and R with fixed aperture angle can be done in $O(n^3 \log n)$ time.*

3.3 Diwedge separability

Two intersecting planes divide the 3D space into four quadrants. The union of a pair of opposite quadrants is called a *diwedge*. The diwedge separability problem is the following: Is there a diwedge that contains R but does not contain any point from B or vice versa?

Theorem 8. *Deciding whether the sets B and R are diwedge separable can be done in $O(n^4)$ time.*

We define the *aperture angle* of a diwedge to be the bigger of the two aperture angles defined by the planes of the diwedge. We consider the problem of computing the separating diwedge with maximum or minimum aperture angle. As per the definition, the aperture angle can take values between 90° and 180° .

Theorem 9. *Computing a separating diwedge for B and R with maximum or minimum aperture angle can be done in $O(n^4 \log n)$ time.*

We also consider the problem of deciding whether there exists a separating diwedge with a fixed aperture angle θ , $90^\circ \leq \theta < 180^\circ$.

Theorem 10. *Computing a separating diwedge for B and R with fixed aperture angle can be done in $O(n^4 \log n)$ time.*

4 Separability by a linear number of planes

In this section we study other separability criteria which can be consider also as extensions for slice, wedge and double wedge separability in the plane, but allowing a linear number of planes. More precisely, we extend the concepts above to prismatic, pyramidal and dipyramidal separability, respectively.

4.1 Prismatic separability

A prism is defined as the space swept by a convex polygon when it is moved along a line perpendicular to its plane; the direction of this line is called the *prism direction*. The prismatic separability problem asks the question: Is there an infinite prism which contains all red points but none of blue points or vice versa?

Theorem 11. *Deciding whether the sets B and R are prismatic separable and computing the locus of the prism directions of all feasible solutions can be done in $O(n^3)$ time. The locus is formed by at most $O(n^2)$ connected components and its total complexity is $O(n^2 \alpha(n))$.*

A still open problem is how can we find a minimum (in number of faces) separating prism? The minimum prism has at least three faces, hence this must address the question of deciding triangular prismatic separability which will be consider in section 5. We can compute the minimum separating prism for a given direction of prism u . This problem is equivalent to computing the minimum (in number of edges) convex polygon which separates the projected red and blue points on a plane with normal vector u ; this problem can be solved in $O(n \log n)$ optimal time [1, 7].

4.2 Pyramidal separability

Join all the vertices of a convex polygon to a point in space to get a pyramid. The convex polygon is called the base of the pyramid while the point in space is called its apex. An infinite pyramid is one whose base is at infinity. The pyramidal separability problem asks the question: Is there an infinite pyramid that contains all the red points but none of the blue points or vice versa?

Theorem 12. *Deciding pyramidal separability for B and R and computing the locus of apices of all the separating pyramids can be done with a randomized algorithm whose expected running time is $O(n^3 \log^2 n)$. The locus of apices of separating pyramids is formed by $O(n^3)$ connected components with $O(n^3 \log n)$ total complexity.*

Next theorem shows a deterministic algorithm for solving the decision problem.

Theorem 13. *Deciding whether the sets B and R are pyramidal separable can be done in $O(n^7)$ time.*

4.3 Dipyramidal separability

Dipyramidal separability has the same definition as pyramidal separability except that now we have two symmetrical pyramids having the same apex. Assume that the red points R are inside the possible separating dipyramid which produces a partition of R .

Theorem 14. *Deciding whether the sets B and R are dipyramidal separable can be done in $O(n^8 \log n)$ time.*

5 Separability by a constant number of planes

In this section we study some particular separability criteria which involve three to six planes. More precisely, we consider triplane, triangular prism, tetrahedral and box separability.

Three intersecting planes divide the 3D space into eight octants. The triplane separability problem asks the question: Are there three planes such that each of the octants they define has points of only one color?

Theorem 15. *Deciding whether the sets B and R are triplane separable can be done in $O(n^7)$ time.*

The triangular prismatic separability problem asks the question: Is there an infinite triangular prism which contains all the red points but none of the blue points or vice versa?

Theorem 16. *Deciding whether the sets B and R are triangular prismatic separable can be done in $O(n^5)$ time.*

The tetrahedral separability problem asks the question: Is there a tetrahedron that contains all the red points but none of the blue points or vice versa?

Theorem 17. *Deciding whether the sets B and R are tetrahedral separable can be done in $O(n^7)$ time.*

The box separability problem asks the question: Is there an orthogonal box which contains all the red points but none of the blue or vice versa?

Theorem 18. *Deciding whether the sets B and R are box separable can be done in $O(n^7)$ time.*

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] B. Aronov, M. Sharir, *The Common Exterior of Convex Polygons in the Plane*, Computational Geometry: Theory and Applications, 8, 1997, pp. 139–149.
- [3] B. Aronov, M. Sharir, B. Tagansky *The Union of Convex Polyhedra in Three Dimensions*, SIAM J. Comput., 26 (6), 1997, pp. 1670–1688.
- [4] B. Chazelle, H. Edelsbrunner, *An Optimal Algorithm for Intersecting Line Segments*, Journal of ACM, 39, 1992, pp. 1–54.
- [5] B. Chazelle, H. Edelsbrunner, L. Guibas, M. Sharir, *Diameter, Width, Closest Line Pair, and Parametric Searching*, 8th Annual Symp. on Comput. Geom., 1992, pp. 120–129.
- [6] G. Davis, *Computing Separating Planes for a Pair of Disjoint Polytopes*, First Annual Symp. on Comput. Geom., 1985, pp. 8–14.
- [7] H. Edelsbrunner, F. P. Preparata, *Minimum Polygonal Separation*, Information and Computation, 77, 1988, pp. 218–232.
- [8] M. E. Houle, *Algorithms for Weak and Wide Separation of Sets*, Discrete Applied Mathematics, Vol. 45, 1993, pp. 139–159.
- [9] M. E. Houle and G. T. Toussaint, *Computing the Width of a Set*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 10, No. 5, 1988, pp. 761–765.
- [10] 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.
- [11] F. Hurtado, M. Mora, P. A. Ramos, C. Seara, *Two Problems on Separability with Lines and Polygonal*, 15th European Workshop on Computational Geometry, 1999, pp. 33–35.
- [12] G. O. Katona, *On a Problem of L. Fejes Tóth*, Stud. Sci. Math. Hung., 12, 1977, pp. 77–80.
- [13] N. Megiddo, *Linear-time Algorithms for Linear Programming in \mathbb{R}^3 and Related Problems*, SIAM J. Comput., 12 (4), 1983, pp. 759–776.
- [14] F. P. Preparata, S. J. Hong, *Convex Hulls of Finite Sets of Points in Two and Three Dimensions*, Comm. of ACM, 20, 1977, pp. 87–93.