Exact Construction of Minimum-Width Annulus of Disks in the Plane*

Ophir Setter[†] Dan Halperin[†]

Abstract

The construction of a minimum-width annulus of a set of objects in the plane has useful applications in diverse fields, such as tolerancing metrology and facility location. We present a novel implementation of an algorithm for obtaining a minimum-width annulus containing a given set of disks in the plane, in case one exists. The algorithm extends previously known methods for constructing minimum-width annuli of sets of points. The algorithm for disks requires the construction of two Voronoi diagrams of different types, one of which we call the "farthest-point farthest-site" Voronoi diagram and appears not to have been investigated before. The vertices of the overlay of these two diagrams are candidates for the annulus' center. The implementation employs an asymptotically near-optimal randomized divide-and-conquer algorithm for constructing two-dimensional Voronoi diagrams. Our software utilizes components from CGAL, the Computational Geometry Algorithms Library, and follows the exact computation paradigm. We do not assume general position. Namely, we handle degenerate input and produce exact results.

1 Introduction

An *annulus* is the bounded area between two concentric circles. The width of an annulus is the difference between the radii of its outer and inner bounding circles. Given a set of objects in the plane, the objective is to find a minimum-width annulus containing those objects. Figure 1(c) shows such an annulus for a set of disks. Constructing a minimum-width annulus has applications in various fields including tolerancing metrology and facility location [11, 19].

A minimum-width annulus does not always exist. If the width of the set of objects[1] is smaller than the width of any containing annulus, then there is no minimum-width annulus.

In the case of point sets, a minimum-width annulus must have (at least) two points on each of its outer and inner circles [15]. Hence, the center of a minimum-width annulus must lie on an intersection point of the nearest-neighbor and the farthest-neighbor Voronoi diagrams of the points. Using this observation, an algorithm for finding a minimum-width annulus of planar points was developed [5, 16].

Similar methods were used to solve different variations of the problem, such as finding a minimum-width annulus of point sets with different constraints on its radii (e.g., fixed inner radius) [3], and finding a minimum-width annulus bounding a polygon [10]. For some special cases there are specific deterministic sub-quadratic algorithms [4, 8, 17].

Agarwal and Sharir introduced the most efficient (randomized) algorithm to date for constructing a minimum-width annulus of planar points, which achieves an expected running time of $O(n^{3/2+\varepsilon})$ [1].

2 Solving the Problem for Disks

Given a set of objects O in the plane (also called *Voronoi sites*) and a distance function ρ, the *nearest-neighbor Voronoi diagram* of O with respect to ρ is the partition of the plane into maximally connected cells, where each cell consists of points that are closer to one particular site (or a set of sites) than to any other site. The *bisector* of two Voronoi sites is the locus of all points that have an equal distance to both sites. A similar definition is used to define the *farthest-neighbor Voronoi diagram*.

Recall that in the case of point sets, a minimum-width annulus can be found by overlaying the nearest-neighbor and farthest-neighbor Voronoi diagrams of the points. We show that a similar approach applies to the case of sets of disks, but the relevant diagrams require more careful definitions.

Instead of constructing the nearest-neighbor Voronoi diagram of the points we construct the *additively-weighted Voronoi diagram* of the disks, also known as the *Apollonius diagram* [14].

The Apollonius diagram is the Voronoi diagram defined for disks with respect to the following distance function ρ. For a point p and a disk D with a center c and radius r, we define $\rho(p, D) = ||p - c|| - r$. The distance between a point outside a disk and the disk is the Euclidean distance. Apollonius bisectors, which compose the diagram, are branches of hyperbolas.

Instead of constructing a farthest-neighbor Voronoi diagram of points we construct a different diagram,

*This work has been supported in part by the Israel Science Foundation (grant no. 236/06), by the German-Israeli Foundation (grant no. 969/07), and by the Hermann Minkowski–Minerva Center for Geometry at Tel Aviv University.

†School of Computer Science, Tel-Aviv University, 69978, Israel. {ophirset,danha}@post.tau.ac.il

[1]The width of a set is defined to be the width of the thinest strip (i.e., the area bounded between two parallel lines) containing it.

which requires the definition of another distance function. Consider the following farthest-point distance function from a point $p \in \mathbb{R}^2$ to a set of points $S \subset \mathbb{R}^2$:

$$\rho(p, S) = \sup_{x \in S} ||p - x||,$$

which measures the farthest distance from the point p to the set S. Consider the farthest-neighbor Voronoi diagram with respect to this distance function. We call this diagram the "farthest-point farthest-site" (FPFS) Voronoi diagram. The distance function ρ becomes the Euclidean distance when the set S consists of a single point. However, this is not the case when the set S is, say, a disk in the plane.

The following lemma characterizes the FPFS Voronoi diagram of disks in the plane, showing that the bisectors induced by its sites are hyperbolic arcs.

Lemma 1 *The bisector of two disks in the plane induced by the farthest-point distance function is one branch of a hyperbola, and is identical to the Apollonius bisector of the disks with swapped radii.*

Proof. Let $(c_A, r_A), (c_B, r_B)$ be two disks in the plane with respective centers c_A, c_B and radii r_A, r_B. Their farthest-point distance bisector is the zero set of the equation $||x - c_A|| + r_A = ||x - c_B|| + r_B$, which is the same as the zero set of $||x - c_A|| - r_B = ||x - c_B|| - r_A$. The latter equation describes the Apollonius bisector of $(c_A, r_B), (c_B, r_A)$. □

We now prove that there is a minimum-width annulus (in case one exists) whose center is a vertex of the overlay of the Apollonius diagram and the FPFS Voronoi diagram of the disks.

Let $\mathcal{D} = \{d_1, \ldots, d_n\}$ be a collection of disks in the plane, such that for all i, $d_i \not\subseteq \bigcup_{j \neq i} d_j$. For simplicity of exposition, we assume here that $n \geq 3$; the case of $n < 3$ is simple to handle. Let $\mathcal{I}_N, \mathcal{O}_N \subseteq \mathcal{D}$ denote the set of disks that touch the inner and outer circles of a bounding annulus N, respectively. We show that there is a minimum-width annulus whose circles intersect the disks of $\mathcal{D}$ in at least 4 points.

Theorem 2 *If there is a minimum-width annulus containing $\mathcal{D}$, then there is a minimum-width annulus N such that $|\mathcal{I}_N| + |\mathcal{O}_N| \geq 4$.*

We omit the detailed proof in this extended abstract. Each minimum-width annulus must touch the disks of $\mathcal{D}$ in at least two points, as we can shrink $\mathcal{O}_N$ and expand $\mathcal{I}_N$ until each of them touches a disk. In the case where N does not touch the disks of $\mathcal{D}$ in at least 4 points, we can move N and obtain a smaller width annulus.

Applying Theorem 2, we distinguish between three cases for a possible location of the center of a minimum-width annulus:

1. $|\mathcal{I}_N| \geq 3$ and $|\mathcal{O}_N| = 1$ – the center coincides with a vertex of the Apollonius diagram.
2. $|\mathcal{I}_N| = 1$ and $|\mathcal{O}_N| \geq 3$ – the center coincides with a vertex of the FPFS Voronoi diagram.
3. $|\mathcal{I}_N| \geq 2$ and $|\mathcal{O}_N| \geq 2$ – the center lies on an intersection point of the Apollonius diagram and the FPFS Voronoi diagram.

We therefore construct each of the diagrams and overlay them. For each vertex of the overlay, we retrieve four relevant disks (either three touching the inner circle and the one touching the outer circle, in case 1, or three touching the outer circle and the one touching the inner circle, in case 2, or two pairs of disks touching respectively the inner and outer circles), and compute the width of the resulting annulus. We output the annulus of the smallest width. Figure 1 illustrates the algorithm for computing a minimum-width annulus of a set of disks and a highly degenerate input, which is handled properly by our implementation.

The FPFS Voronoi diagram can be defined as a farthest site abstract Voronoi diagram [12]. Hence, FPFS Voronoi diagrams are of linear complexity in the size of the input (as are Apollonius diagrams).

3 Constructing General Voronoi Diagrams with CGAL

As described in Section 2 above, the process of constructing a minimum-width annulus bounding a set of disks mainly comprises three geometric operations: (i) construction of the Apollonius diagram, (ii) construction of the FPFS Voronoi diagram, and (iii) overlay of the two diagrams. This section describes some of the software components which our implementation is based on, their various ramifications on the algorithm, and how they work in synergy to yield an exact and robust implementation, which can handle input that is not necessarily in general position and produce results of arbitrary precision.

The connection between Voronoi diagrams and envelopes is long-known [6], and yields a useful approach for constructing various types of Voronoi diagrams. CGAL,[2] the Computational Geometry Algorithms Library, contains a robust and efficient implementation of a divide-and-conquer algorithm for constructing envelopes of general surfaces in 3-space [13]. This implementation was employed to yield a general framework for constructing two-dimensional Voronoi diagrams [9].

Like most algorithms and data-structures in CGAL, the framework follows the generic programming paradigm, and is independent of the type of input sites. The generic implementation is parameterized with a *traits* class that must provide all required types and methods to construct and handle bisector curves

[2] `http://www.cgal.org`

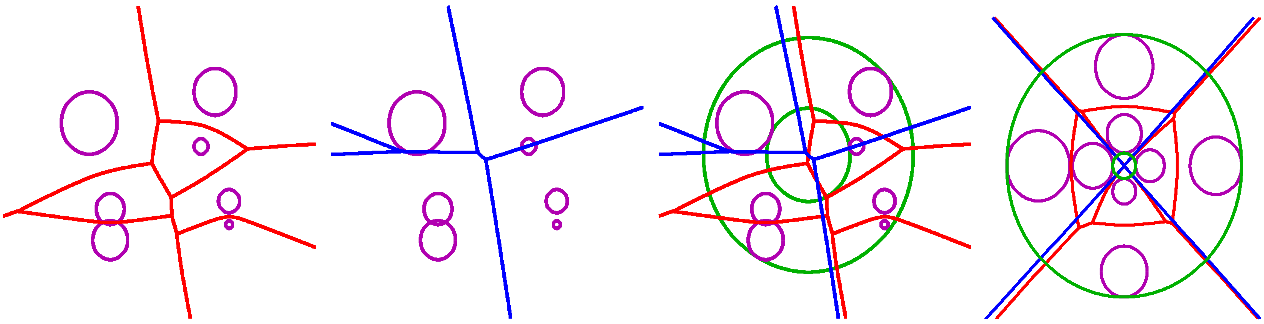

Figure 1: Constructing a minimum-width annulus of a set of disks. (a) The Apollonius diagram of the set of disks. (b) The FPFS Voronoi diagram of the set of disks. (c) A minimum-width annulus of the set of disks. The center of the annulus is a vertex in the overlay of the Apollonius and the FPFS Voronoi diagrams. (d) A highly degenerate scenario for constructing a minimum-width annulus of a set of disks.

(e.g., intersecting bisector curves, comparing the y-coordinates of a point and its vertical projection on a curve), and to answer proximity queries (see Section 4).

The decision to use the aforementioned framework to construct both Voronoi diagrams required in the minimum-width annulus algorithm has two major advantages. First, the framework is bisector-based; namely, the main basic operation it requires is the construction of bisectors of Voronoi sites. This simplifies the implementation of the diagrams as both have the same bisector type (Lemma 1). Second, the resulting diagrams are represented as CGAL arrangements, which can be passed as input to consecutive operations supported by CGAL. One of those operations is a sweep-based overlay [18] that is used to carry out the next step of our algorithm.

A straightforward application of the divide-and-conquer approach for Voronoi diagrams yields algorithms with worst-case running time of $O(n^{2+\varepsilon})$ even for diagrams of linear complexity. Through randomization, it has been shown that the expected running time is lower:

Theorem 3 [9] *For a specific type of two-dimensional Voronoi diagrams, so that the worst-case complexity of the diagram of any set of at most n sites is $O(n)$, the divide-and-conquer envelope algorithm computes it in expected $O(n \log^2 n)$ time.*

In our case, applying Theorem 3 yields an expected construction time of $O(n \log^2 n)$ for both the Apollonius and the FPFS Voronoi diagrams using the divide-and-conquer algorithm. Overlaying the two diagrams with the sweep-based algorithm has $O((n + k) \log n)$ worst-case time complexity where k is the number of intersections between the diagrams. The total expected running time of the algorithm is therefore $O(n \log^2 n + k \log n)$, where k can be $\Theta(n^2)$.

Though the worst-case complexity of the algorithm is super-quadratic, it is reasonable to assume that the expected time complexity is, in many cases, smaller. Indeed, we may have $\Theta(n^2)$ intersections between the two diagrams, but, though not proven for disks, for random point sets it is known that the expected number of intersections between the farthest and the nearest Voronoi diagrams is linear [2].

4 Implementation Details

Following the description of the framework for Voronoi diagrams, we have implemented two traits classes for constructing Apollonius diagrams and FPFS Voronoi diagrams named `Algebraic_apollonius_traits_2` and `Algebraic_farthest_point_farthest_site_traits_2`, respectively.

Both traits classes are based on the algebraic plane curves traits for the arrangement package [7], which provides all the algebraic functionality needed to handle algebraic bisector curves in a robust and exact manner. The remaining functors required by the framework are functors for constructing the bisectors of two disks (which consist of selecting the correct branch of a hyperbola) and functors to answer different proximity queries.

Each required proximity predicate is given a set of points P in the plane (e.g., an edge) and two Voronoi sites, and should indicate which of the sites is closer to P. All required proximity predicates are implemented similarly as follows: we construct a point p inside P using the algebraic infrastructure, and then answer the proximity query by comparing the Apollonius (or the farthest-point) distances from p to the sites. We expedite the comparisons here, as well as the comparisons of widths of annuli later, by using rational interval arithmetic, alleviating the need for the heavier exact algebraic machinery wherever possible.

As both the Apollonius and the FPFS Voronoi diagrams have the same bisectors type and similar distance functions (Lemma 1), both `Algebraic_apollonius_traits_2` and `Algebraic_farthest_point_farthest_site_traits_2` inherit from the same base class, and define only a distance function and a functor for bisectors' construction (`Algebraic_farthest_point_farthest_site_traits_2` swaps the radii of the disks and then calls the same function as `Algebraic_apollonius_traits_2`), maximizing code reuse.

We use the generic overlay function from CGAL's arrangement package to overlay the Apollonius diagram and the FPFS Voronoi diagram. The overlay function is parameterized with an overlay-traits class, which updates the resulting arrangement's features based on data associated with the input arrangements' features [18]. We have created a specially-tailored overlay-traits class for updating the features of the overlay with sites from both diagrams.

Disks	Time	V	E	F
50	10.741	126	213	88
100	26.994	238	395	158
200	62.968	416	659	244
500	185.244	775	1174	400
1000	405.405	1242	1894	653

On the left you can see the time consumption (in seconds) of the algorithm execution on different input sizes, as well as the size of the final overlay (the vertices of which are the candidates for the center of the annulus). The experiments were carried out on an Intel® Core$^{\text{TM}}$2 Duo 2.00GHz processor with 1GB memory.

5 Future Work

It would be interesting to further investigate the farthest-point farthest-site Voronoi diagram, presented above, for disk sites in particular and for other objects in general, and find additional applications for this type of diagrams.

Another direction to pursue is the application of this approach to objects other than points [5, 16], polygons [10], or disks, as described in this paper.

Acknowledgments

The authors are grateful to all those who have supported this work by advice, discussion, or fruitful cooperation. Special thanks are given to Efi Fogel, Eric Berberich, and Barak Raveh.

References

[1] P. K. Agarwal and M. Sharir. Efficient randomized algorithms for some geometric optimization problems. *Disc. Comput. Geom.*, 16:317–337, 1996.

[2] P. Bose and L. Devroye. Intersections with random geometric objects. *Comput. Geom. Theory Appl.*, 10(3):139–154, June 1998.

[3] M. de Berg, P. Bose, D. Bremner, S. Ramaswami, and G. T. Wilfong. Computing constrained minimum-width annuli of point sets. *Computer-Aided Design*, 30(4):267–275, 1998.

[4] O. Devillers and P. A. Ramos. Computing roundness is easy if the set is almost round. *Int. J. Comput. Geom. Appl.*, 12:229–248, 2002.

[5] H. Ebara, N. Fukuyama, H. Nakano, and Y. Nakanishi. Roundness algorithms using the Voronoi diagrams. In *Proc. 16th The Canadian Conf. on Comput. Geom.*, volume 41, 1989.

[6] H. Edelsbrunner and R. Seidel. Voronoi diagrams and arrangements. *Disc. Comput. Geom.*, 1:25–44, 1986.

[7] A. Eigenwillig and M. Kerber. Exact and efficient 2D-arrangements of arbitrary algebraic curves. In *Proc. 19th Annu. ACM-SIAM Symp. Disc. Alg.*, pages 122–131. Soc. for Industrial and Applied Math., 2008.

[8] J. Garcia-Lopez, P. A. Ramos, and J. Snoeyink. Fitting a set of points by a circle. *Disc. Comput. Geom.*, 20(3):389–402, 1998.

[9] D. Halperin, O. Setter, and M. Sharir. Constructing two-dimensional voronoi diagrams via divide-and-conquer of envelopes in space. ACS technical report ACS-TR-361601-01, TAU, 2008.

[10] V.-B. Le and D.-T. Lee. Out-of-roundness problem revisited. *IEEE Trans. Pattern Anal. Mach. Intell.*, 13(3):217–223, 1991.

[11] M. T. Marsh and D. A. Schilling. Equity measurement in facility location analysis: A review and framework. *European Journal of Operational Research*, 74(1):1–17, April 1994.

[12] K. Mehlhorn, S. Meiser, and R. Rasch. Furthest site abstract Voronoi diagrams. *Int. J. of Comput. Geom. Appl.*, 11(6):583–616, 2001.

[13] M. Meyerovitch. Robust, generic and efficient construction of envelopes of surfaces in three-dimensional space. In *Proc. 14th Annu. Eur. Symp. Alg.*, pages 792–803, 2006.

[14] A. Okabe, B. Boots, K. Sugihara, and S. N. Chiu. *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams.* Wiley, NYC, 2nd edition, 2000.

[15] T. J. Rivlin. Approximating by circles. *Computing*, 21:93–104, 1979.

[16] U. Roy and X. Zhang. Establishment of a pair of concentric circles with the minimum radial separation for assessing roundness error. *Computer-Aided Design*, 24(3):161–168, 1992.

[17] K. Swanson, D.-T. Lee, and V. L. Wu. An optimal algorithm for roundness determination on convex polygons. In *Comput. Geom. Theory Appl.*, pages 601–609, 1993.

[18] R. Wein, E. Fogel, B. Zukerman, and D. Halperin. Advanced programming techniques applied to CGAL's arrangement package. *Comput. Geom. Theory Appl.*, 38(1–2):37–63, 2007. Special issue on CGAL.

[19] C. K. Yap. Exact computational geometry and tolerancing metrology. Technical Report SOCS-94.50, McGill School of Computer Science, 1994.