Voronoi diagrams and Delaunay triangulations

Steven Fortune

July 19, 1995

1 Introduction

The Voronoi diagram of a set of sites partitions space into regions, one per site; the region for a site s consists of all points closer to s than to any other site. The dual of the Voronoi diagram, the Delaunay triangulation, is the unique triangulation so that the circumsphere of every triangle contains no sites in its interior. Voronoi diagrams and Delaunay triangulations have been rediscovered or applied in many areas of mathematics and the natural sciences; they are central topics in computational geometry, with hundreds of papers discussing algorithms and extensions.

Section 2 discusses the definition and basic properties in the usual case of point sites in R^d with the Euclidean metric, while section 3 gives basic algorithms. Some of the many extensions obtained by varying metric, sites, environment, and constraints are discussed in section 4. Section 5 finishes with some interesting and nonobvious structural properties of Voronoi diagrams and Delaunay triangulations.

Glossary

site: A defining object for a Voronoi diagram or Delaunay triangulation. Also generator, source, Voronoi point.

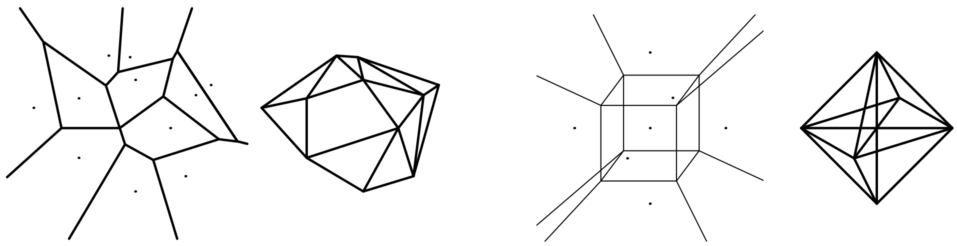
Voronoi face: The set of points for which a single site is closest (or more generally a set of sites is closest). Also Voronoi region, Voronoi cell.

Voronoi diagram: The set of all Voronoi faces. Also Thiessen diagram, Wigner-Seitz diagram, Blum transform, Dirichlet tesselation.

Delaunay triangulation: The unique triangulation of a set of sites so that the circumsphere of each triangle has no sites in its interior.

2 Point sites in the Euclidean metric

See [5, 13, 15] for more details and proofs of material in this section.



(a) (d) (c) (d)

Figure 1: Voronoi diagram and Delaunay triangulation of the same set of sites in two dimensions(a,b) and three dimensions(c,d).

Definitions Let S be a finite set of points in R^d, called *sites*. The *Voronoi face* of a site $s \in S$ is the set of all points of R^d strictly closer to s than to any other site in S. The Voronoi face of a site is always a nonempty, open, convex, full-dimensional subset of R^d. More generally, for T a nonempty subset of S, the *Voronoi face* $V(T)$ is the set of points of R^d equidistant from all members of T and closer to any member of T than to any member of $S \backslash T$. The *Voronoi diagram* of S is the collection of all nonempty Voronoi faces $V(T)$, for $T \subseteq S$. The Voronoi diagram forms a cell complex partitioning R^d.

In two dimensions (figure 1(a)), the Voronoi face of a site is the interior of a convex, possibly infinite polygon; its boundary consists of *Voronoi edges* (1-dimensional faces) equidistant from two sites and *Voronoi vertices* (0-dimensional faces) equidistant from at least three sites. Figure 1(c) shows a Voronoi diagram in three dimensions.

For T a subset of S, a *Delaunay face* $D(T)$ is defined whenever there is a sphere through all the sites of T with all other sites exterior (equivalently, whenever $V(T)$ is nonempty); then $D(T)$ is the (relative) interior of the convex hull of T. For example, in two dimensions (figure 1(b)), a *Delaunay triangle* is formed by three sites whose circumcircle is empty and a *Delaunay edge* connects two sites that have an empty circumcircle (in fact, infinitely many empty circumcircles). The *Delaunay triangulation* of S is the collection of all Delaunay faces. The Delaunay triangulation forms a cell complex partitioning the convex hull of S.

There is an obvious one-one correspondence between the Voronoi diagram and the Delaunay triangulation; it maps the Voronoi face $V(T)$ to the Delaunay face $D(T)$. This correspondence has the property that the sum of the dimensions of $V(T)$ and $D(T)$ is always d. Thus in two dimensions $V(T)$ is a Voronoi vertex iff $D(T)$ is an open polygonal region; $V(T)$ is an edge iff $D(T)$ is; $V(T)$ is an open polygonal region iff $D(T)$ is a vertex, i.e. a site. In fact, the one-one correspondence is a duality between cell complexes, reversing face ordering: for subsets $T, T' \subseteq S$, $V(T')$ is a face of $V(T)$ iff $D(T)$ is a face of $D(T')$.

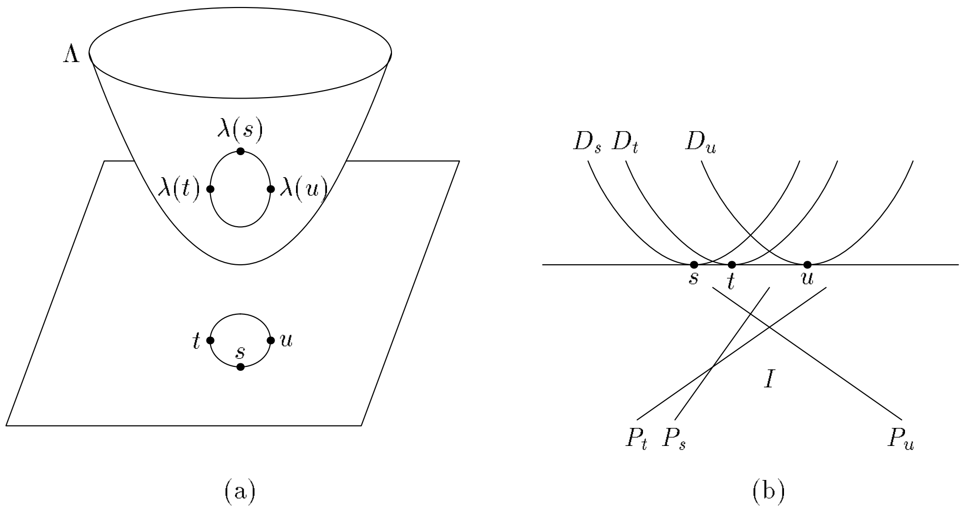


Figure 2: (a) The intersection of a plane with Λ is an ellipse that projects to a circle; (b) on any vertical line, the surfaces $\{D_s\}$ appear in the same order as the planes $\{P_s\}$.

The set of sites $S \subset R^d$ is in *general position* (or is *nondegenerate*) if no $d+2$ points lie on a common d-sphere and no $k+2$ points lie on a common k-flat, for $k<d$. If S is in general position, then the Delaunay triangulation of S is a simplicial complex, and every vertex of the Voronoi diagram has degree $d+1$. If S is not in general position, then Delaunay faces need not be simplices; for example the four cocircular sites in figure 1(b) form a Delaunay quadrilateral. A *completion* of a Delaunay triangulation is obtained by splitting nonsimplicial faces into simplices without adding new vertices.

Relation to convexity There is an intimate connection between Delaunay triangulations in R^d and convex hulls in R^{d+1}, and between Voronoi diagrams in R^d and halfspace intersections in R^{d+1}. To see the connections, consider the special case of d=2. Identify R^2 with the plane spanned by the first two coordinate axes of R^3, and call the third coordinate direction the *vertical* direction.

The *lifting map* $\lambda : R^2 \rightarrow R^3$ is defined by $\lambda(x_1, x_2) = (x_1, x_2, x_1^2 + x_2^2)$; $\Lambda = \lambda(R^2)$ is a paraboloid of revolution about the vertical axis. See figure 2(a). Let H be the convex hull of the lifted sites $\lambda(S)$.

The Delaunay triangulation of S is exactly the orthogonal projection into R^2 of the lower faces of H (a face is *lower* if it has a supporting plane with inward normal having positive vertical coordinate). To see this informally, suppose that triangle $\lambda(s)\lambda(t)\lambda(u)$ is a lower facet of H, and that plane P passes through $\lambda(s)\lambda(t)\lambda(u)$. The intersection of P with Λ is an ellipse that projects orthogonally to a circle in R^2 (figure 2(a)). Since all other lifted sites are above the plane, all other unlifted sites are outside the circle, and stu is a Delaunay triangle. The opposite direction, that a Delaunay triangle is a lower facet, is similar.

For Voronoi diagrams, assign to each site $s = (s_1, s_2)$ the plane

$$P_s = \{(x_1, x_2, x_3) : x_3 = -2x_1 s_1 + s_1^2 - 2x_2 + s_2^2\}.$$

Let I be the intersection of the lower halfspaces of the planes P_s. The Voronoi diagram is exactly the orthogonal projection into R^2 of the upper faces of I. To see this informally, consider the surfaces

$$D_s = \{(x_1, x_2, x_3) : x_3 = ((x_1 - s_1)^2 + (x_2 - s_2)^2\}$$

(see figure 2). Viewed as a function from R^2 into R, D_s gives the squared distance to site s. Furthermore, P_s and D_s differ only by the quadratic term $x_1^2 + x_2^2$, which is independent of s. Hence a point $x \in R^2$ is in the Voronoi cell of site t iff on the vertical line through x, D_t is lowest among all surfaces $\{D_s\}$. This happens exactly if on the same line, P_t is lowest among all planes $\{P_s\}$, i.e. x is in the projection of the upper face of I formed by P_t.

Combinatorial complexity In dimension 2, a Delaunay triangulation of $n \geq 3$ sites has at most $2n-5$ triangles and $3n-6$ edges (and the Voronoi diagram has at most as many vertices and edges, respectively). In dimension $d \geq 3$ the Delaunay triangulation can have $\Theta(n^{\lceil d/2 \rceil})$ faces. For a lower bound example in dimension 3, choose $n/2$ distinct point sites on each of two noncoplanar line segments l and l'. Then there is an empty sphere through each quadruple of sites (a, a', b, b') with a, a' adjacent on l and b, b' adjacent on l'. Since there are $\Omega(n^2)$ such quadruples, there are as many Delaunay tetrahedra. Exact worst-case bounds are known.

If point sites are chosen uniformly at random from inside a sphere, then the expected number of faces is linear in the number of sites. In dimension 2, the expected number of triangles is $2n$; in dimension 3, the expected number of tetrahedra is $\sim 6.77n$; in dimension 4, the expected number of 4-simplices is $\sim 31.78n$ [12]. Similar bounds probably hold for other distributions, but proofs are lacking.

3 Basic algorithms

Figure 3 lists basic algorithms that compute the Delaunay triangulation of n point sites in R^d using the Euclidean metric. Using the connection with convexity, any d+1-dimensional convex hull algorithm can be used to compute a d-dimensional Delaunay triangulation; in fact the divide-and-conquer, incremental, and gift-wrapping algorithms are specialized convex hull algorithms. Running times are given both for worst-case inputs, and for inputs chosen uniformly at random inside a sphere, with expectation taken over input distribution. The Voronoi diagram can be obtained in linear time from the Delaunay triangulation, using the one-one correspondence between their faces. See [5, 13, 15, 16] for more citations.

algorithm	d	worst case	uniform	citation
flipping	2	$O(n^2)$		Chapter XXX
plane sweep	2	$O(n \log n)$		
divide-and-conquer	2	$O(n \log n)$	$O(n)$	[17, 19]
randomized-incremental	2	$O(n \log n)$		
randomized-incremental	≥ 3	$O(n^{\lceil d/2 \rceil})$	$O(n \log n)$	
gift-wrapping	≥ 2	$O(n^{\lceil d/2 \rceil + 1})$	$O(n)$	[12]

Figure 3: Delaunay triangulation algorithms in the Euclidean metric for point sites.

The randomized-incremental algorithm The incremental algorithm adds sites one by one, updating the Delaunay triangulation after each addition. The update consists of discovering all Delaunay faces whose circumspheres contain the new site. These faces are deleted and the empty region is partitioned into new faces, each of which has the new site as a vertex. See Figure 4. An efficient algorithm requires a good data structure for finding the faces to be deleted. Then the running time is determined by the total number of face updates, which depends upon site insertion order. The bounds given in figure 3 are the expected running time of an algorithm that makes a random choice of insertion order, with each insertion permutation equally likely; the bounds for the worst-case insertion order are about a factor of n worse. (For uniform data there is a double expectation, over both insertion order and input distribution). With additional algorithmic complexity, it is possible to obtain deterministic algorithms with the same worst-case running times[9].

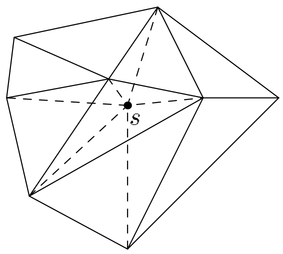


Figure 4: The addition of site s deletes four triangles and adds six (shown dashed).

The plane sweep algorithm The plane sweep algorithm computes a planar Delaunay triangulation using a horizontal line that sweeps upwards across the plane. The algorithm discovers a Delaunay triangle when the sweepline passes through the topmost point of its circumcircle; in figure 5, the Delaunay triangles shown in black have already been discovered. A sweepline data structure stores an ordered list of sites; the entry for site s corresponds to an interval I_s on the sweepline where each maximal empty circle with topmost point in I_s touches site s. The sweepline moves in discrete steps only when the order changes. This happens when a new site is encountered or when a new Delaunay triangle is discovered (at the topmost point of the circumcircle of three sites that are consecutive on the sweepline list). A priority queue is needed to determine the next sweepline move. The running time of the algorithm is $O(n \log n)$ since the sweepline moves $O(n)$ times–once per site and once per triangle–and it costs time $O(\log n)$ per move to maintain the priority queue and sweepline data structure.

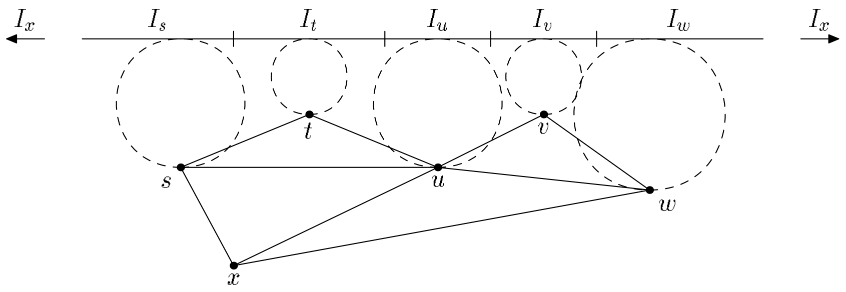


Figure 5: The sweepline list is x, s, t, u, v, w, x. The next Delaunay triangle is tuv.

Other algorithms The divide-and-conquer algorithm uses a splitting line to partition the point set into two equal halves, recursively computes the Delaunay triangulation of each half, and then merges the two subtriangulations in linear time. The gift-wrapping algorithm is a specialization of the convex-hull gift-wrapping algorithm (Chapter XXX) to Delaunay triangulations.

Output-sensitive algorithms, with running time approximately proportional to the actual number of Delaunay facets, have remained elusive. Recent progress is in [8].

If the sites form the vertices of a convex polygon, then the Voronoi diagram can be computed in linear time[1].

Implementations Many of these algorithms have been implemented; the World Wide Web site `http://www.geom.umn.edu/locate/cglib` has pointers to publicly available code.

4 Extensions

Higher-order Voronoi diagrams The *order-k* Voronoi diagram partitions R^d on the basis of the first k closest sites (without distinguishing order among them). The *furthest site* Voronoi diagram partitions R^d on the basis of the furthest site, or equivalently, the closest $n-1$ of n sites. The order-k Voronoi diagram can be obtained as an appropriate projection of the k-level of an arrangement of hyperplanes[13, 16]; it can also be obtained as the orthogonal projection of an intersection polytope[7]. In dimension 2, the order-k Voronoi diagram has $O(k(n-k))$ faces. In dimensions $d \geq 3$, the sum of the number of faces of the order-j diagrams, $j \leq k$, is $O(n^{\lceil d/2 \rceil} k^{\lfloor d/2 \rfloor + 1})$[11]; good bounds for fixed k remain an open problem.

problem	d	time	citation
furthest site	2	$O(n \log n)$	
furthest site	≥ 3	$O(n^{\lceil d/2 \rceil})$	
order-k	2	$O(k(n-k) \log n + n \log^3 n)$	[2]
order-j, $1 \leq j \leq k$	≥ 3	$O(n^{\lceil d/2 \rceil} k^{\lfloor d/2 \rfloor + 1})$	[23]

Figure 6: Algorithms for order-k Voronoi diagrams of point sites in the Euclidean metric.

Visibility constraints Let S be a set of n point sites in R^2 and E a set of noncrossing *constraint* edges with endpoints in S. A point $p \in R^2$ is *visible* from a site s if the open segment ps does not intersect any edge of E. A *constrained Delaunay triangulation* (CDT) of S using E is a triangulation of S extending the edges in E so that the circumcircle of every triangle contains no site that is visible from all three sites defining the triangle. The CDT is as close as possible to the true Delaunay triangulation, subject to the constraint that the edges in E must be used. See also Chapter XXX on mesh generation.

The *bounded distance* from a site to a point is Euclidean distance if the point is visible, and infinite otherwise; the *bounded Voronoi diagram* of S using E is defined using bounded distance. The bounded Voronoi diagram is dual to a subgraph of the CDT.

Both the CDT and the bounded Voronoi diagram can be computed in time $O(n \log n)$ using either divide-and-conquer or sweepline paradigm. If the sites and constraint edges are the vertices and edges of a simple polygon, respectively, then the CDT can be computed in linear time [20].

There is no obvious generalization of constrained Delaunay triangulations to dimension $d \geq 3$, since there exist polyhedra in R^3 that cannot be triangulated, at least without using Steiner points.

Other distance measures Figure 7 lists Voronoi diagram algorithms where 'distance' is altered. The distance from a site s_i to a point x can be a function of the euclidean distance $e(s_i, x)$ and a site-specific real weight w_i.

The seemingly-peculiar *power distance* is the distance from x to the sphere of radius $\sqrt{w_i}$ about s_i along a line tangent to the sphere. Many of the basic Voronoi diagram algorithms extend immediately to the power distance, even in higher dimension.

A *(polygonal) convex distance function* is defined by a convex polygon C with the origin in its interior. The distance from x to y is the real $r \geq 0$ so that the boundary

problem	distance to x	time	citation
additive weights	$w_i + e(s_i, x)$	$O(n \log n)$	
multiplicative weights	$w_i e(s_i, x)$	$O(n^2)$	[6]
Laguerre or power	$(e(s_i, x)^2 - w_i)^{1/2}$	$O(n \log n)$	[4]
L_p	$\|\|s_i - x\|\|_p$	$O(n \log n)$	[22]
convex distance function		$O(n \log n)$	[10]
abstract	axiomatic	$O(n \log n)$	[21]
simple polygon	geodesic	$O(n \log^2 n)$	[3]
crystal growth	$w_i \cdot SP(s_i, x)$	$O(n^3 + nS \log S)$	[28]

Figure 7: Algorithms for point sites in dimension 2, varying distance measure.

of $rC+x$ contains y. Polygonal convex distance functions generalize the L_1 and L_∞ metrics (C is a diamond or square, respectively); a polygonal convex distance function is a metric exactly if C is symmetric about the origin.

An *abstract* Voronoi diagram is defined by the 'bisectors' between pairs of sites, which must satisfy special properties.

The *geodesic* distance inside an environment of polygonal obstacles is the length of the shortest path that avoids obstacle interiors. Recent progress using the geodesic metric appears in [18].

The *crystal growth* Voronoi diagram models crystal growth where each crystal has a different growth rate. The distance from a site s_i to a point x in the Voronoi face of s_i is $w_i \cdot SP(s_i, x)$, where w_i is a weight and $SP(s_i, x)$ is the shortest path distance lying entirely within the Voronoi face of s_i. The parameter S in the running time measures the time to approximate bisectors numerically.

Other sites The Voronoi diagram of a set of n line segment sites can be computed in time $O(n \log n)$ using the sweepline method or the divide-and-conquer method; the medial axis of a polygon or polygonal region can be obtained from the Voronoi diagram of its constituent line segments. The divide-and-conquer algorithm extends to circular-arc segments as well. The Voronoi diagram of a set of circles can be computed using an additively-weighted point-site algorithm.

Motion planning The motion planning problem is to find a collision-free path for a robot in an environment filled with obstacles. The Voronoi diagram of the obstacles is quite useful, since it gives a lower-dimensional skeleton of maximal clearance from the obstacles. In many cases the shape of the robot can be used to define an appropriate metric for the Voronoi diagram. See Chapter XXX on motion planning.

5 Important properties

Roundness The Delaunay triangulation is "round", that is, skinny simplices are avoided. This can be formalized in two dimensions by Lawson's classic result: over all possible triangulations, the Delaunay triangulation maximizes the minimum angle of any triangle. No generalization using angles is known in higher dimension. However, define the *enclosing radius* of a simplex as the minimum radius of an enclosing sphere. In any dimension and over all possible triangulations of a point set, the Delaunay triangulation minimizes the maximum enclosing radius of any simplex[26]. Also see Chapter XXX on mesh generation.

Visibility depth ordering Choose a viewpoint v and a family of disjoint convex objects in R^d. Object A is *in front of* object B from v if there is a ray starting at v that intersects A and then B in that order. Though an arbitrary family can have cycles in the "in front of" relation, the relation is acyclic for the faces of the Delaunay triangulation, for any viewpoint and any dimension[14].

An application comes from computer graphics. The "painter's algorithm" renders three-dimensional objects in back to front order, with later objects simply overpainting the image space occupied by earlier objects. A valid rendering order always exists if the "in front of" relation is acyclic, as is the case if the objects are Delaunay tetrahedra, or a subset of a set of Delaunay tetrahedra.

Subgraph relationships The edges of a Delaunay triangulation form a graph whose vertices are the set of sites. In any dimension, the following subgraph relations hold:

$$EMST \subseteq RNG \subseteq GG \subseteq DT$$

where EMST is the Euclidean minimum spanning tree, RNG is the relative neighborhood graph, and GG is the Gabriel graph. See also Chapter XXX on pattern recognition.

Dilation A geometrically embedded graph G has *dilation* c if for any two vertices, the shortest path distance along the edges of G is at most c times the Euclidean distance between the vertices. In R^2, the edge set of the Delaunay triangulation has dilation at most ~ 2.42; with an equilateral-triangle convex distance function, the dilation is at most 2.

Interpolation Suppose each point site $s_i \in S \subset R^d$ has an associated function value f_i. For $p \in R^d$ define $\lambda_i(p)$ as the proportion of the area of s_i's Voronoi cell that would be removed if p were added as a site. Then the *natural neighbor* interpolant $f(p) = \sum \lambda_i(p) f_i$ is C^0, and C^1 except at sites. A more complex C^1 interpolant can be obtained as well[29].

Alternatively, for a triangulation of S in R^2, consider the piecewise linear surface defined by linear interpolation over each triangle. Over all possible triangulations, the Delaunay triangulation minimizes the roughness of the resulting surface, where *roughness* is the L_2 norm squared of the gradient of the surface, integrated over the triangulation[27].

6 Further reading

Survey papers by Aurenhammer[5] and Fortune[15] cover many aspects of Delaunay triangulations and Voronoi diagrams. The book by Okabe, Boots, and Sugihara[24] is entirely devoted to Voronoi diagrams, and has an extensive discussion of applications. Basic reference for geometric algorithms are [13, 25].

References

[1] A. Aggarwal, L.J. Guibas, J. Saxe, P.W. Shor, A linear-time algorithm for computing the Voronoi diagram of a convex polygon, *Disc. Comp. Geom* 4:591-604, 1989.

[2] P. K. Agarwal, M. de Berg, J. Matoušek, O. Schwarzkopf, Constructing levels in arrangments and higher order Voronoi diagrams, *Proc. Tenth Ann. Symp. Comp. Geom* pp. 67–75, 1994.

[3] B. Aronov, On the geodesic Voronoi diagram of point sites in a simple polygon, *Algorithmica* 4:109–140, 1989.

[4] F. Aurenhammer, Power diagrams: properties, algorithms, and applications, *SIAM J. Comp.* 16(1):78–96, 1987.

[5] F. Aurenhammer, Voronoi diagrams – a survey of a fundamental geometric data structure, *ACM Comp. Surveys* 23:345–405, 1991.

[6] F. Aurenhammer, H. Edelsbrunner, An optimal algorithm for constructing the weighted Voronoi diagram in the plane. *Pattern Recognition* 17:251–257, 1984.

[7] F. Aurenhammer, O. Schwarzkopf, A simple randomized incremental algorithm for computing higher order Voronoi diagrams, *Proc. Seventh Ann. Symp. Comp. Geom*, pp. 142–151, 1995.

[8] T. M. Y. Chan, J. Snoeyink, C.K. Yap, Output-sensitive construction of polytopes in four dimensions and clipped Voronoi diagrams in three, *Proc. Sixth. Ann. Symp. Disc. Alg.*, pp. 282–291, 1995.

[9] B. Chazelle, An optimal convex hull algorithm and new results on cuttings, *Proc. 32nd Ann. Symp. Found. Comp. Sci.* pp. 29–38, 1991.

[10] L. P. Chew, R. L. Drysdale, Voronoi diagrams based on convex distance functions, *Proc. 1st Ann. Symp. Comp. Geom* pp. 235-244, 1985.

[11] K.L. Clarkson, P. W. Shor, Applications of random sampling in computational geometry, II. *Disc. Comp. Geom.* 4:387–421, 1989.

[12] R. Dwyer, Higher-dimensional Voronoi diagrams in linear expected time, *Disc. Comp. Geom* 6:343–367 1991.

[13] H. Edelsbrunner, *Algorithms in Combinatorial Geometry*, Springer-Verlag, Berlin, 1987.

[14] H. Edelsbrunner, An acyclicity theorem for cell complexes in d dimensions, *Combinatorica* 10:251–260, 1990.

[15] S. Fortune, Voronoi diagrams and Delaunay triangulations, *Computing in Euclidean Geometry*, D.Z. Du, F. Hwang, eds, pp. 193–233, World Scientific, 1992.

[16] S. Fortune, Progress in computational geometry, *Directions in geometric computing*, R. Martin, ed., pp. 81–128, Information Geometers, Ltd, 1993.

[17] L.J. Guibas, J. Stolfi, Primitives for the manipulation of general subdivisions and the computation of Voronoi diagrams, *ACM Trans. Graphics* 4(2):74–123, 1985.

[18] J. Hershberger, S. Suri, Efficient computation of Euclidean shortest paths in the plane, *34th Ann. Symp. Found. Comp. Sci.* pp. 508–517, 1993.

[19] J. Katajainen, Bucketing and filtering in computational geometry, Ph.D. thesis, University of Turku, Finland, 1987.

[20] R. Klein, A. Lingas, A linear-time randomized algorithm for the bounded Voronoi diagram of a simple polygon, *Proc. Ninth Ann. Symp. Comp. Geom*, pp. 124–132, 1993.

[21] R. Klein, K. Mehlhorn, S. Meiser, Randomized incremental construction of abstract Voronoi diagrams, *Comp. Geom. Theory and Appl.* 3(3):157–184, 1993.

[22] D. T. Lee, Two-dimensional Voronoi diagrams in the L_p metric, *JACM* 27(4): 604–618, 1980.

[23] K. Mulmuley, *Computational Geometry: an introduction through randomized algorithms*, Prentice-Hall, 1994.

[24] A. Okabe, B. Boots, K. Sugihara, *Spatial Tesselations: concepts and applications of Voronoi diagrams*, Wiley Series in Probability and Mathematical Statistics, Wiley, 1992.

[25] F. P. Preparata, M.I. Shamos, *Computational geometry*, Springer-Verlag, 1985.

[26] V. T. Rajan, Optimality of the Delaunay triangulation in R^d, *Disc. Comp. Geom.* 12(2):189–202, 1994.

[27] S. Rippa, Minimal roughness property of the Delaunay triangulation, *Comp. Aided. Design* 7:489–497, 1990.

[28] B. Schaudt, R. L. Drysdale, Multiplicatively weighted crystal growth Voronoi diagram, *Proc. 7th. Ann. Symp. Comp. Geom.*, pp. 214–223, 1991.

[29] R. Sibson, Natural neighbor interpolation, *Interpreting Multivariate Data*, V. Barnett, ed., pp. 21–36, Wiley, 1981.