Randomized incremental construction of the Hausdorff Voronoi diagram of non-crossing clusters*

Panagiotis Cheilaris† Elena Khramtcova† Evanthia Papadopoulou†

Abstract

We give a randomized incremental construction for the Hausdorff Voronoi diagram of non-crossing clusters of points. Our best complexity algorithm takes expected time $O(n \log^2 n (\log\log n)^2)$ and worst-case space $O(n)$, improving upon previous results. A simpler-to-implement algorithm, based on the Voronoi hierarchy, is also given, which takes expected time $O(n \log^3 n)$ and expected space $O(n)$. To achieve this, we augment the Voronoi hierarchy with the ability to efficiently handle non-standard characteristics of generalized Voronoi diagrams, such as sites of non-constant complexity, sites that are not enclosed in their Voronoi regions, and empty Voronoi regions.

1 Introduction

Given a family F of point clusters in $\mathbb{R}^2$, the *Hausdorff Voronoi diagram* of F is a subdivision of the plane into regions such that the *Hausdorff Voronoi region* of a cluster $P \in F$ is the locus of points closer to P than to any other cluster in F. The closeness of a point $t \in \mathbb{R}^2$ to a cluster P is measured by the *farthest distance* $d_f(t, P) = \max_{p \in P} d(t, p)$, where $d(t, p)$ is the Euclidean distance between t and p. The motivation for investigating Hausdorff Voronoi diagrams comes from its use in efficiently estimating the sensitivity of a Very Large Scale Integration (VLSI) design to random defects during manufacturing [11].

Clusters P and Q are called *non-crossing* if the convex hull of $P \cup Q$ admits at most two supporting segments with one endpoint in P and one endpoint in Q (see Figure 1). We assume that clusters are pairwise non-crossing, unless stated otherwise.

The Hausdorff Voronoi diagram of non-crossing clusters has size $O(n)$ [12], where n is the total number of points in all clusters. It can be constructed in time $O(n^2)$ [8], or in time $O((n + K) \log n)$ [11, 12], however, K can be superlinear.[1] It is an instance of abstract Voronoi diagrams [10], thus it can also be constructed in expected time $O(bn \log n)$, where $b = O(n)$ is the time to construct the bisector between two clusters [1]. A more recent algorithm gives $O(n \log^4 n)$ time and $O(n \log^2 n)$ space complexity [6].

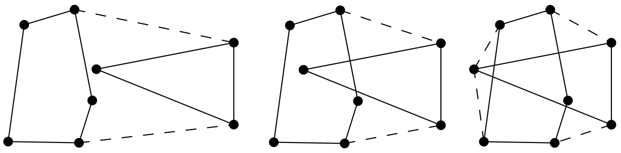

Figure 1: disjoint, non-crossing, crossing clusters.

In this work, we build the Hausdorff Voronoi diagram using a *randomized incremental* approach. That is, sites (clusters) are inserted one by one in random order and the diagram is updated at every insertion [5]. The bottleneck in this approach is to identify fast a point $t \in \mathbb{R}^2$ that will lie in the region of the new site. This is difficult for the Hausdorff Voronoi diagram because: (a) the region of the new cluster (site) might not contain any of its points, (b) the insertion of the new cluster can make an existing region empty, and (c) clusters have non-constant size, and thus the computation of a bisector or the answering of an *in-circle test* require non-constant time. To overcome these issues we exploit properties of Hausdorff Voronoi diagram and we maintain a *dynamic point location* data structure, which is also used to perform simple parametric search queries.[2] Our approach is modular as it can use any dynamic point location data structure. If we use the data structure by Baumgarten *et al.* [2], then we get an algorithm which takes expected time $O(n \log^2 n (\log\log n)^2)$ and uses linear space. Alternatively, if we augment the *Voronoi hierarchy* [9] with the ability to efficiently handle the difficulties (a) to (c), we obtain a more practical algorithm which takes expected $O(n \log^3 n)$ time and $O(n)$ space.

The augmentation of the Voronoi hierarchy may be of interest for incremental construction of other generalized Voronoi diagrams.

*Supported in part by the Swiss National Science Foundation grant 134355, under the auspices of the ESF EUROCORES program EuroGIGA/VORONOI.

†Faculty of Informatics, Università della Svizzera italiana, Lugano, Switzerland

[1] K is the number of pairs of clusters such that one cluster is contained in a specially defined enclosing circle of the other, e.g., the minimum enclosing circle [12].

[2] Construction algorithms for the Hausdorff Voronoi diagram [6] and for the farthest-polygon Voronoi diagram [4] also resort to parametric search.

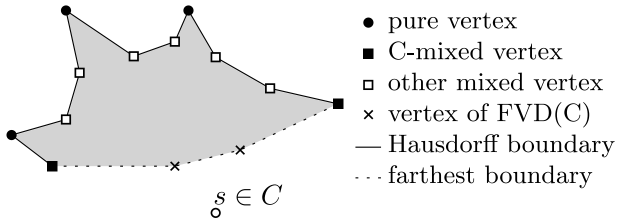

Figure 2: $\mathrm{hreg}(s), s \in C$.

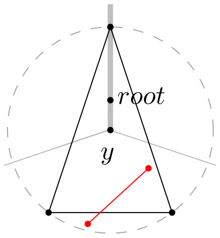

Figure 3: 3-point cluster C (black), and 2-point cluster P (red) limiting w.r.t. $y \in \mathrm{fskel}(P)$.

2 Definitions and Structural properties

Let $F = \{C_1, \ldots, C_m\}$ be a family of non-crossing clusters of points such that no two clusters have a common point. For simplicity we assume that no four points lie on the same circle. Let $\operatorname{conv} P$ denote the convex hull of cluster P and $\mathrm{CH}(P)$ denote the sequence of points of P on the boundary of $\operatorname{conv} P$, in counterclockwise order. For $s \in C$, the *farthest region* of s in the *farthest Voronoi diagram* (FVD) of C is:

$$\mathrm{freg}_C(s) = \{p \mid \forall s' \neq s\colon d(p,s) > d(p,s')\}.$$

The graph structure of the FVD of C, $|C| > 1$, forms a tree, called the *farthest skeleton* of C, $\mathrm{fskel}(C)$. If $|C| = 1$ then $\mathrm{fskel}(C)$ is C itself.

The *Hausdorff region* of a cluster $C \in F$ and a point $s \in C$ are defined as

$$\mathrm{hreg}_F(C) = \{p \mid \forall C' \neq C\colon d_{\mathrm{f}}(p,C) < d_{\mathrm{f}}(p,C')\};$$
$$\mathrm{hreg}_F(s) = \mathrm{hreg}_F(C) \cap \mathrm{freg}_C(s).$$

The boundary of the Hausdorff region of a point $s \in C$ consists of two chains: (1) the *farthest boundary* of s, which is the portion of $\mathrm{fskel}(C)$ in $\mathrm{hreg}_F(C)$, i.e., $\mathrm{bd}\,\mathrm{hreg}_F(s) \cap \mathrm{bd}\,\mathrm{freg}_C(s)$; (2) the *Hausdorff boundary* of s, i.e., $\mathrm{bd}\,\mathrm{hreg}_F(s) \cap \mathrm{bd}\,\mathrm{hreg}_F(C)$. Neither chain can be empty if $\mathrm{hreg}_F(C) \neq \emptyset$ and $|C| > 1$.

As shown in [12], there are three types of vertices on the boundary of a Hausdorff Voronoi region $\mathrm{hreg}_F(s)$: (1) Standard Voronoi vertices, which are equidistant from C and two other clusters, referred to as *pure* vertices (using the terminology of [4]). (2) *Mixed* vertices, which are equidistant from C and one other cluster. The mixed vertices which are equidistant to two points of C and one point of another cluster are called *C-mixed* vertices; there are exactly two of them on the boundary of $\mathrm{hreg}_F(s)$. (3) Vertices of $\mathrm{fskel}(C)$ on the farthest boundary of s. See Figure 2.

Useful properties of the Hausdorff Voronoi diagram are summarized in Proposition 1. We need some definitions.

Consider a cluster C. Line segment $\overline{ab}$ is a *chord* of C if $a, b \in \mathrm{CH}(C)$. Assign a root in $\mathrm{fskel}(C)$ arbitrarily and denote this rooted tree by $T(C)$. Let y be any point of $\mathrm{fskel}(C)$, and $\overline{cc^*}$ be a chord of C such that y lies on the bisector between c and c^*. Let D_y be the closed disk centered at y with radius $d_{\mathrm{f}}(y,C)$. Then, $C \subset D_y$. Point y subdivides $\mathrm{fskel}(C)$ into two parts. Denote the part containing the descendants of y in $T(C)$ by $T(y)$, and its complement by $T^{\sim}(y)$. Chord $\overline{cc^*}$ subdivides D_y into D^{r}_y and D^{f}_y, where D^{r}_y (resp., D^{f}_y) is the *rear (forward) part*, enclosing the portion of $\operatorname{conv} C$ inducing $T(y)$ ($T^{\sim}(y)$). See Figure 3.

Definition 1 *Cluster P is* limiting *with respect to point $y \in \mathrm{fskel}(C)$, if disk D_y contains P. Cluster P is called* forward limiting *if $P \subset D^{\mathrm{f}}_y \cup \operatorname{conv} C$ or* rear limiting *if $P \subset D^{\mathrm{r}}_y \cup \operatorname{conv} C$.*

The following properties are derived directly from [12].

Proposition 1 *Let F be a family of non-crossing clusters and $C, P, Q \in F$. Then:*

(i) If $\mathrm{hreg}_F(C) \neq \emptyset$, then $\mathrm{hreg}_F(C) \cap \mathrm{fskel}(C)$ is non-empty and connected.

(ii) Let y be a point of $\mathrm{fskel}(C)$ such that y is closer to cluster P than to C. If P is forward (resp. rear) limiting with respect to y then the entire $T(y)$ (resp. $T^{\sim}(y)$) is closer to P than to C.

(iii) Let $\overline{uv}$ be an edge of $\mathrm{fskel}(C)$. If both u and v are closer to P than to C then $\mathrm{hreg}_F(C)$ does not intersect $\overline{uv}$.

(iv) Region $\mathrm{hreg}_F(C) = \emptyset$ if and only if either there is a cluster $P \subset \operatorname{conv} C$, or there exists a pair of clusters $\{P, Q\}$ such that P is rear limiting and Q is forward limiting with respect to the same point $y \in \mathrm{fskel}(P)$. Pair $\{P, Q\}$ is called a killing pair *for C.*

3 General incremental construction algorithm

Let $C_1, C_2, \ldots, C_m$ be a fixed order of clusters. Let F_i denote the family of the first i clusters according to this order. The incremental approach constructs successively the Hausdorff Voronoi diagram of F_1, F_2, $\ldots$, $F_m = F$. For each cluster C_i, we have the farthest Voronoi diagram, $\mathrm{FVD}(C_i)$, and a (static) point location data structure on $\mathrm{FVD}(C_i)$.

We construct $\mathrm{HVD}(F_{i+1})$ from $\mathrm{HVD}(F_i)$ by inserting C_{i+1}. We first find a point t, which is closer to C_{i+1} than to any cluster in F_i, or if there is no such point, we conclude that $\mathrm{hreg}_{F_{i+1}}(C_{i+1}) = \emptyset$ and stop. By Proposition 1(i), it is sufficient to search for

t just in $\mathrm{fskel}(C_{i+1})$. Then, we trace the region of C_{i+1} around t and update the diagram.

We first consider vertices of $\mathrm{fskel}(C_{i+1})$. For each such vertex w:

- Find the nearest to w cluster $C^w \in F_i$ using point location in $\mathrm{HVD}(F_i)$,
- if $d_\mathrm{f}(w, C_{i+1}) < d_\mathrm{f}(w, C^w)$, then $t = w$; exit the procedure. Else, if possible, eliminate from further consideration a subtree of $\mathrm{fskel}(C_{i+1})$ incident to w; see Proposition 1(ii).

If no vertex is found, then consider any remaining edges of $\mathrm{fskel}(C_{i+1})$ as $\mathrm{hreg}_{F_{i+1}}(C_{i+1})$ may intersect the interior of at most one edge of $\mathrm{fskel}(C_{i+1})$; see Proposition 1(i).

Edge uv of $\mathrm{fskel}(C_{i+1})$ is called a *candidate* edge if $d_\mathrm{f}(u, C_{i+1}) < d_\mathrm{f}(u, C^v)$ and $d_\mathrm{f}(v, C_{i+1}) < d_\mathrm{f}(v, C^u)$. By Proposition 1(iii), it is sufficient to only check a candidate edge and there can be at most one such edge for C_{i+1}. Thus, for a candidate edge (if any) we perform parametric search to decide whether $\mathrm{hreg}_{F_{i+1}}(C_{i+1})$ is empty or not, and still to find a point t in this region in the latter case.

Lemma 2 *Suppose clusters are inserted in a uniformly random order. Then, the expected time complexity of the randomized algorithm is*

$$O(n \log n) + O(n)(t_\mathrm{q}(n) + t_\mathrm{i}(n) + t_\mathrm{d}(n)) + m \cdot t_\mathrm{p}(n),$$

where $t_\mathrm{q}(n)$, $t_\mathrm{p}(n)$, $t_\mathrm{i}(n)$, $t_\mathrm{d}(n)$ are the times for a query, a parametric search, an insertion, and a deletion[3] *in a point location data structure for a diagram of complexity $O(n)$, respectively.*

Proof. (Sketch) The expected total number of insertions and deletions is $O(n)$.[4] The total time for the construction of farthest Voronoi diagrams for all clusters is $O(n \log n)$. For each cluster C_i, we perform $O(|C_i|)$ point location queries and at most one parametric search. Thus, in total we perform $O(n)$ point locations and at most m parametric searches. □

We can use the dynamic point location data structure of Baumgarten *et al.* [2] with $t_\mathrm{q}(n) = O(\log n \log\log n)$, $t_\mathrm{i}(n) = O(\log n \log\log n)$, and $t_\mathrm{d}(n) = O(\log^2 n)$. Parametric search can be performed as a simulation of a point location query for the unknown point t in time $t_\mathrm{p}(n) = (t_\mathrm{q}(n))^2$ (see also [4]). As a result:

Theorem 1 *There is a randomized algorithm that constructs the Hausdorff Voronoi diagram of a family of non-crossing clusters in linear space and in expected time $O(n \log^2 n (\log\log n)^2)$.*

4 Augmenting the Voronoi hierarchy

The Voronoi hierarchy [9] is a simple randomized point location data structure for Voronoi diagrams inspired from the Delaunay hierarchy [7]. For a family F of general sites, each level ℓ of the hierarchy corresponds to a subset $F^{(\ell)}$ of F and stores the Voronoi diagram of $F^{(\ell)}$. Level 0 corresponds to F. A *Voronoi hierarchy* of *height* k is then: $F = F^{(0)} \supseteq F^{(1)} \supseteq \ldots \supseteq F^{(k)}$. For all $\ell \in \{1, \ldots, k\}$, $F^{(\ell)}$ is a random sample of $F^{(\ell-1)}$ according to a Bernoulli distribution with parameter $\beta \in (0, 1)$. The expected height of the hierarchy for a family of m sites is $O(\log m)$. *Point location* in the Voronoi hierarchy for a query point q works as follows. Starting from the topmost level k, for each level ℓ, find the site S^ℓ in $F^{(\ell)}$ which is the nearest to the query point q, by performing a *walk*. Each step of the walk reduces the distance to q from the current site S by moving to a site, neighboring to S. Walk at level $\ell - 1$ starts from S^ℓ. The answer to the query is S^0.

For the Hausdorff Voronoi diagram several complications arise: (a) Sites are of non-constant complexity. (b) We need to perform parametric search i.e., a walk for an unknown point along a candidate edge. (c) Voronoi regions might be empty.

To address (a) we need the concept of an *active* point. Consider a cluster C at level ℓ, such that $\mathrm{hreg}_{F^{(\ell)}}(C) \neq \emptyset$. For brevity, let this region be denoted as $\mathrm{hreg}^{(\ell)}_F(C)$, and similarly for the regions of individual points.

Definition 2 *Point $c \in C$ is* active *at level ℓ, if $\mathrm{hreg}^{(\ell)}_F(c) \neq \emptyset$. The set $\hat{C}^{(\ell)}$ of all active points of C at level ℓ is called the* active set *of C at level ℓ. For brevity, $d^{(\ell)}_\mathrm{f}(t, C) = d_\mathrm{f}(t, \hat{C}^{(\ell)})$.*

Performing one step of the walk. Let C be the current cluster visited during the walk at level ℓ, and q be the query point. The next cluster C' in the walk is determined as follows. Let c be a point of $\hat{C}^{(\ell)}$ such that $q \in \mathrm{freg}_{\hat{C}^{(\ell)}}(c)$. Let $v_1, \ldots, v_j$ be a list of pure vertices on the Hausdorff boundary of $\mathrm{hreg}^{(\ell)}_F(c)$, in counterclockwise order, and let $C^0, \ldots, C^j, C^{j+1}$ be respective adjacent clusters. The rays $\overrightarrow{cv_1}, \ldots, \overrightarrow{cv_j}$ partition $\mathrm{freg}_{\hat{C}^{(\ell)}}(c)$ into $j + 1$ unbounded regions. If $\overrightarrow{cq}$ is just after the ray $\overrightarrow{cv_i}$ or just before the ray $\overrightarrow{cv_{i+1}}$, then set $C' = C^i$.

In order to find C' in $O(\log n)$ time, for each cluster C at level ℓ such that $C \in F^{(\ell)}$ we store the binary trees containing: (1) the active set $\hat{C}^{(\ell)}$; (2) for each $c \in \hat{C}^{(\ell)}$, the list of all pure vertices adjacent to $\mathrm{hreg}^{(\ell)}_F(c)$ (see Figure 2).

The parametric search. For $\ell \in \{0, \ldots, k\}$, let I^ℓ be the interval of points on uv which are closer to C_{i+1}

[3] The time for insertion and deletion can be amortized.

[4] It can be shown with the Clarkson-Shor technique [5].

than to any cluster in $F_i^{(\ell)}$. (By convention, $I^{k+1} = uv$.) Then, $uv = I^{k+1} \supseteq I^k \supseteq I^{k-1} \supseteq \cdots \supseteq I^1 \supseteq I^0$. If $I^\ell \neq \emptyset$, we compute the leftmost endpoint u^ℓ of interval I^ℓ, i.e., the endpoint which is closer to u. If $I^0 \neq \emptyset$, then u^0 is the answer to the query.

From the point $u^{\ell+1}$ and the cluster of $F_i^{(\ell+1)}$ closest to $u^{\ell+1}$, point u^ℓ is computed entirely at level ℓ. We find a sequence of points $u^{\ell+1} = a_0, a_1, \ldots, a_r = u^\ell$. For each point a_j, we keep track of the cluster C^{a_j} in F_i^ℓ which is the closest to a_j. We compute a_{j+1} from a_j as follows.

- If $d_{\mathrm{f}}(a_j, C_{i+1}) \leq d_{\mathrm{f}}(a_j, C^{a_j})$, we set $u^\ell = a_j$ and continue to the next level.
- Else, if $d_{\mathrm{f}}(v, C^{a_j}) \leq d_{\mathrm{f}}(v, C_{i+1})$, we stop and report that $\mathrm{hreg}_{F_{i+1}}(C_{i+1}) = \emptyset$.
- Otherwise, we determine a_{j+1} by a (standard) parametric search in $\mathrm{FVD}(C^{a_j})$ with segment $a_j v$. Then, we perform a walk (at level ℓ) from cluster C^{a_j} to find the cluster $C^{a_{j+1}}$ closest to a_{j+1}. If $C^{a_{j+1}} = C^{a_j}$, we set $u^\ell = a_j$ and continue to the next level.

Empty Voronoi regions. When C_{i+1} is inserted, an existing non-empty region of a cluster P may become empty. If P has an empty region at level ℓ, but a non-empty region at level $\ell + 1$, then the point location for a query point $q \in \mathrm{hreg}^{(\ell+1)}_{F_{i+1}}(P)$ will give an error. To fix the problem, we link cluster P at level $\ell + 1$ to at most two other clusters at level ℓ (see Proposition 1(iv)), so that every point $q \in \mathbb{R}^2$ is strictly closer to either one of them than to P, as follows.

While inserting C_{i+1} at level ℓ, we keep track of the list V of all the (deleted) P-mixed vertices.

At level $\ell + 1$, for each P-mixed vertex v, we check if $d_{\mathrm{f}}(v, C_{i+1}) \geq d_{\mathrm{f}}(v, P)$. If yes, we store the point $c \in C_{i+1}$ for which $d_{\mathrm{f}}(v, C_{i+1}) = d(v, c)$.

1. If all P-mixed vertices are closer to C_{i+1} than to P, we link P only to C_{i+1}.
2. Else, we link P to its killing pair $\{K, C_{i+1}\}$, such that $K \in F^{(l)}$; see Proposition 1(iv).

We identify cluster K using the list V and the point c. Each vertex $u \in V$ is equidistant from points p_u, $p_u^* \in P$, and $q_u \in Q$, for some $Q \in F_i^{(\ell)}$. We check whether c and q_u are on different sides of the chord $\overline{p_u p_u^*}$. If yes, then set $K = Q$ and stop.

The complexity is analyzed in the following.

Lemma 3 *The expected length of the walk at level ℓ is constant.*

Lemma 4 *A point location query and a parametric search query are answered in expected $O(\log^2 n)$ and $O(\log^3 n)$ time respectively.*

Lemma 5 *Let n be the sum of the sizes of all sites in a family of sites F. Assuming that the underlying type of Voronoi diagram for F is of size $O(n)$, the expected size of the Voronoi hierarchy for F is also $O(n)$.*

Theorem 2 *The Hausdorff Voronoi diagram of non-crossing clusters can be constructed in $O(n \log^3 n)$ expected time and $O(n)$ expected space, using the Voronoi hierarchy.*

For details, see [3].

References

[1] M. Abellanas, G. Hernandez, R. Klein, V. Neumann-Lara, and J. Urrutia. A combinatorial property of convex sets. *Discrete Comput. Geom.*, 17(3):307–318, 1997.

[2] H. Baumgarten, H. Jung, and K. Mehlhorn. Dynamic point location in general subdivisions. *J. Algorithms.*, 17(3):342–380, 1994.

[3] P. Cheilaris, E. Khramtcova, and E. Papadopoulou. Randomized incremental construction of the Hausdorff Voronoi diagram of non-crossing clusters. Technical Report 2012/03, University of Lugano, 2012.

[4] O. Cheong, H. Everett, M. Glisse, J. Gudmundsson, S. Hornus, S. Lazard, M. Lee, and H.-S. Na. Farthest-polygon Voronoi diagrams. *Comput. Geom.*, 44(4):234–247, 2011.

[5] K. Clarkson and P. Shor. Applications of random sampling in computational geometry, II. *Discrete Comput. Geom.*, 4:387–421, 1989.

[6] F. Dehne, A. Maheshwari, and R. Taylor. A coarse grained parallel algorithm for Hausdorff Voronoi diagrams. In *Proc. 35th Int. Conf. on Parallel Processing (ICPP)*, pages 497–504, 2006.

[7] O. Devillers. The Delaunay Hierarchy. *Int. J. Foundations Comput. Sci.*, 13:163–180, 2002.

[8] H. Edelsbrunner, L. J. Guibas, and M. Sharir. The upper envelope of piecewise linear functions: algorithms and applications. *Discrete Comput. Geom.*, 4:311–336, 1989.

[9] M. Karavelas and M. Yvinec. The Voronoi diagram of convex objects in the plane. Technical report RR-5023, INRIA, 2003.

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

[11] E. Papadopoulou. The Hausdorff Voronoi diagram of point clusters in the plane. *Algorithmica.*, 40(2):63–82, 2004.

[12] E. Papadopoulou and D. T. Lee. The Hausdorff Voronoi diagram of polygonal objects: a divide and conquer approach. *Int. J. Comput. Geom. Appl.*, 14(6):421–452, 2004.