A Randomized Divide and Conquer Algorithm for Higher-Order Abstract Voronoi Diagrams

Cecilia Bohler[1], Chih-Hung Liu[1], Evanthia Papadopoulou[2(✉)], and Maksym Zavershynskyi[2]

[1] Institute of Computer Science I, University of Bonn, 53113 Bonn, Germany
bohler@cs.uni-bonn.de, chliu@uni-bonn.de
[2] Faculty of Informatics, Università della Svizzera italiana (USI), Lugano, Switzerland
{evanthia.papadopoulou,maksym.zavershynskyi}@usi.ch

Abstract. Given a set of sites in the plane, their order-k Voronoi diagram partitions the plane into regions such that all points within one region have the same k nearest sites. The order-k abstract Voronoi diagram is defined in terms of bisecting curves satisfying some simple combinatorial properties, rather than the geometric notions of sites and distance, and it represents a wide class of order-k concrete Voronoi diagrams. In this paper we develop a randomized divide-and-conquer algorithm to compute the order-k abstract Voronoi diagram in expected $O(kn^{1+\varepsilon})$ operations. For solving small sub-instances in the divide-and-conquer process, we also give two sub-algorithms with expected $O(k^2 n \log n)$ and $O(n^2 2^{\alpha(n)} \log n)$ time, respectively. This directly implies an $O(kn^{1+\varepsilon})$-time algorithm for several concrete order-k instances such as points in any convex distance, disjoint line segments and convex polygons of constant size in the L_p norm, and others.

Keywords: Higher-Order Voronoi Diagram · Abstract Voronoi Diagram · Randomized Algorithm · Divide and Conquer

1 Introduction

Given a set S of n geometric sites in the plane, their order-k *Voronoi diagram*, $V_k(S)$, is a subdivision of the plane such that every point within an order-k *Voronoi region* has the same k nearest sites. The common boundary between two adjacent Voronoi regions is a *Voronoi edge*, and the common vertex incident to more than two Voronoi regions is a *Voronoi vertex*. The ordinary Voronoi diagram is the order-1 Voronoi diagram, and the farthest-site Voronoi diagram is the order-$(n-1)$ Voronoi diagram.

This work was supported by the European Science Foundation (ESF) in the EUROCORES collaborative research project EuroGIGA/VORONOI, projects DFG Kl 655/17-1 and SNF 20GG21-134355. The work of the last two authors was also supported by the Swiss National Science Foundation, project 200020-149658.

H.-K. Ahn and C.-S. Shin (Eds.): ISAAC 2014, LNCS 8889, pp. 27–37, 2014.
DOI: 10.1007/978-3-319-13075-0_3

For point sites in the Euclidean metric, the order-k Voronoi diagram has been well-studied. Lee [14] showed its structural complexity to be $O(k(n-k))$, and proposed an $O(k^2 n \log n)$-time iterative algorithm. Based on the notions of arrangements and geometric duality, Chazelle and Edelsbrunner [6] developed an algorithm with $O(n^2 + k(n-k) \log^2 n)$ time complexity. Clarkson [7] developed an $O(kn^{1+\varepsilon})$-time randomized divide-and-conquer algorithm, and Agarwal et al. [1], Chan [5], and Ramos [18] proposed randomized incremental algorithms with $O(k(n-k) \log n + n \log^3 n)$, $O(n \log n + nk \log k)$, and $O(n \log n + nk 2^{O(\log^* k)})$ time complexities, respectively. Besides, Boissonnat et al. [4] and Aurenhammer and Schwarzkopf [2] also studied on-line algorithms.

Surprisingly, order-k Voronoi diagrams of sites other than points were only recently considered [17] illustrating different properties from their counterparts for points. For simple, even disjoint, line segments, a single order-k Voronoi region may consist of $\Omega(n)$ disjoint faces; nevertheless, the overall structural complexity of the diagram for n non-crossing line segments remains $O(k(n-k))$ [17]. Abstract Voronoi diagrams were introduced by Klein [10] as a unifying concept to many instances of concrete Voronoi diagrams. They are defined in terms of a system of bisecting curves $\mathcal{J} = \{J(p,q) \mid p,q \in S, p \neq q\}$ rather than concrete geometric sites and distance measures. Order-k abstract Voronoi diagrams were recently considered in [3], providing a unified concept to order-k Voronoi diagrams, and showing the number of their faces to be $\leq 2k(n-k)$. No algorithms for their construction have been available so far. For non-point sites, such as line segments, only $O(k^2 n \log n)$-time algorithms have been available based on the iterative construction [17] and plane sweep [19]. Other recent works on order-k Voronoi diagrams of point-sites in generalized metrics include the L_1/L_∞ metric [16], the city metric [8], and the geodesic order-k Voronoi diagram [15].

In abstract Voronoi diagrams [10], the system of bisecting curves satisfies axioms (A1)–(A5), given below, for any $S' \subseteq S$. Once a concrete bisector system is shown to satisfy these axioms, combinatorial properties and algorithms to construct abstract Voronoi diagrams (see e.g., [10]) are directly applicable. A bisector $J(p,q)$ partitions the plane into two domains $D(p,q)$ and $D(q,p)$, where $D(p,q)$ are points closer to p than q; a first-order Voronoi region $\mathrm{VR}_1(\{p\}, S)$ is defined as $\bigcap_{q \in S, q \neq p} D(p,q)$.

(A1). Each first-order Voronoi region is pathwise connected.
(A2). Each point in the plane belongs to the closure of some first-order Voronoi region.
(A3). No first-order Voronoi region is empty.
(A4). Each curve $J(p,q)$, where $p \neq q$, is unbounded. After stereographic projection to the sphere, it can be completed to be a closed Jordan curve through the north pole.
(A5). Any two curves $J(p,q)$ and $J(s,t)$ have only finitely many intersection points, and these intersections are transversal.

In this paper, we develop a randomized divide-and-conquer algorithm to compute the order-k abstract Voronoi diagram in expected $O(kn^{1+\varepsilon})$ basic operations, based on Clarkson's random sampling technique and one additional axiom:

(A6). The number of vertical tangencies of a bisector is $O(1)$.

Our algorithm is applicable to a variety of concrete order-k Voronoi diagrams satisfying axioms (A1)-(A6), such as point sites in any convex distance metric or the Karlsruhe metric, disjoint line segments and disjoint convex polygons of constant size in the L_p norms, or under the Hausdorff metric. In these instances, all basic operations (see Section 2) can be performed in $O(1)$ time, thus, our algorithm runs in expected $O(kn^{1+\varepsilon})$ time. For non-point sites, this is the first algorithm that achieves time complexity different from the standard $O(k^2 n \log n)$, which is efficient for only small values of k. For point sites in the Euclidean metric, near-optimal randomized algorithms exist [1],[5],[7],[18]; however, they are based on powerful geometric transformations, which are non-trivial to convert to different geometric objects, and/or to the abstract setting, which is based on topological (non-geometric) properties. Matching the time complexity of these algorithms in the abstract setting or for concrete non-point instances remains an open problem.

In order to apply Clarkson's technique [7], we define a vertical decomposition of the order-k Voronoi diagram. We prove that our vertical trapezoidal decomposition allows a divide-and-conquer algorithm and an expected time analysis. When the problem sub-instances are small enough, we propose two sub-algorithms. The first one combines the standard iterative approach [14] and the randomized incremental construction for the order-1 abstract Voronoi diagram [12] and computes the order-k abstract Voronoi diagram in expected $O(k^2 n \log n)$ operations. For the second one, we adopt Har-Peled's method [9] and obtain an $O(n^2 2^{\alpha(n)} \log n)$-operation randomized algorithm, where $\alpha(\cdot)$ is the inverse of the Ackermann function. Our algorithm follows the essence of Clarkson's randomized divide-and-conquer algorithm for the Euclidean order-k Voronoi diagram [7], however, it bypasses all geometric transformations and constraints. Instead, our algorithm defines sub-structures and conflict relations relying on the properties of a bisector system that satisfies the six axioms (A1)–(A6).

2 Preliminaries

Axioms (A1)-(A5) imply that for a given bisecting system $\mathcal{J}$ and a fixed point $x \in R^2$ we can define a linear order on the sites in S.

Definition 1. *For a point $x \in \mathbb{R}^2$ and two sites $p, q \in S$, $p <_x q$, $p =_x q$, or $p >_x q$ if $x \in D(p,q)$, $x \in J(p,q)$, or $x \in D(q,p)$, respectively.*

Since $D(p,q) \cap D(q,r) \subseteq D(p,r)$ [10,11], we can define an ordered sequence on S, $\pi_x^S = (s_1, \ldots, s_n)$, given x, satisfying $s_1 \leq_x s_2 \leq_x \ldots \leq_x s_n$. We say that site s is k-nearest to point x if s occupies the k-th position in the sequence π_x^S.

Definition 2. [3] *The order-k Voronoi region associated with H is*

$$VR_k(H,S) = \bigcap_{p\in H, q\in S\setminus H} D(p,q).$$

The order-k Voronoi diagram is

$$V_k(S) = \bigcup_{|H|=k} \partial VR_k(H,S),$$

where ∂ denotes the boundary.

For each point $x \in \mathrm{VR}_k(H,S)$ and $\pi_x^S = (s_1,\ldots,s_n)$, $H = \{s_1,\ldots,s_k\}$, and $s_k <_x s_{k+1}$. If $\mathrm{VR}_k(H_1,S)$ and $\mathrm{VR}_k(H_2,S)$ share an edge e, then for any point $x \in e$, $H_1 \cap H_2 = \{s_1,\ldots,s_{k-1}\}$ and $s_{k-1} <_x s_k =_x s_{k+1}$, see [3, Lemma 5]. For simplicity, throughout this paper, we make a general position assumption that the degree of any Voronoi vertex is exactly three.

Definition 3. *Let v be a Voronoi vertex among $VR_k(H_1,S)$, $VR_k(H_2,S)$, and $VR_k(H_3,S)$, and let $H = H_1 \cap H_2 \cap H_3$ then v can be categorized into two types:* new *when $|H| = k-1$ and* old *when $|H| = k-2$.*

A *new* Voronoi vertex of $V_k(S)$ is an *old* Voronoi vertex of $V_{k+1}(S)$.

Let v be a Voronoi vertex as in Def. 3. Then we can show that $H = \{s_1,\ldots,s_t\}$ and $s_t <_v s_{t+1} =_v s_{t+2} =_v s_{t+3} <_v s_{t+4}$, where $t = |H|$ and $\pi_v^S = (s_1,\ldots,s_n)$. Each Voronoi vertex is defined by the three sites $s_{t+1}, s_{t+2}, s_{t+3}$.

Definition 4. *The* k-neighborhood *of a site p in S, denoted by $VN_k(p,S)$, is the union of closures of $VR_k(H,S)$ for all $H \subset S$, such that $p \in H$ and $|H| = k$, i.e.,*

$$VN_k(p,S) = \bigcup_{p\in H, H\subset S, |H|=k} \overline{VR_k(H,S)},$$

where $\overline{X}$ denotes the topological closure of the set X.

Each edge of $\partial\mathrm{VN}_k(p,S)$ belongs to $J(p,q)$ for a site $q \in S\setminus\{p\}$, and each edge of $V_k(S)$ belongs to $\partial\mathrm{VN}_k(p,S)$ for a site $p \in S$. The latter condition implies

$$V_k(S) = \bigcup_{p\in S} \partial\mathrm{VN}_k(p,S).$$

Unlike order-k Voronoi regions of point-sites, abstract order-k Voronoi regions may be disconnected. In fact one region may disconnect into $\Omega(n)$ disjoint faces, for $k > 1$ (see e.g. [17] for line segments). Nevertheless, the *k-neighborhood* is connected, and this is the major property used in Section 5.

Lemma 1. *$VN_k(p,S)$ is simply connected and there is no finite set of points whose removal would make $VN_k(p,S)$ disconnected.*

Proof. First we show that $\mathrm{VN}_k(p,S)$ is path connected. The definiton of $\mathrm{VN}_k(p,S)$ implies that p is at most k-nearest for every point in $\mathrm{VN}_k(p,S)$. Therefore $\mathrm{VN}_k(p,S) = \bigcup_{p\in H, H\subset S, |H|=k} \overline{\mathrm{VR}_1(p,\{p\}\cup(S\setminus H))}$. $\mathrm{VR}_1(p,\{p\}\cup(S\setminus H))$ is path connected, axiom (A1). Thus the connectivity of $\mathrm{VN}_k(p,S)$ follows.

Next we show that there can be no holes in $\mathrm{VN}_k(p,S)$. Suppose there is a face F entirely surrounded by $\mathrm{VN}_k(p,S)$. Then all edges on the boundary of F are subsets of $\partial\mathrm{VN}_k(p,S)$. Let the edges correspond to the bisectors $J(p,q_i)$, $i=1,\ldots,m$. If one of the bisectors $J(p,q_i)$ goes through the interior of F then consider a face of $F\cap D(q_i,p)$, which is not empty, and so on until we have a face F' bounded by edges $J(p,q'_1),\ldots,J(p,q'_{m'})$ and $F'\subset D(q'_1,p)\cap\cdots\cap D(q'_{m'},p)$. This implies that F' is a bounded face of the farthest Voronoi region of p in $\{p,q'_1,\ldots,q'_{m'}\}$, a contradiction [3, Lemma 7]. □

Our algorithm, to be described in the sequel, assumes the availability of the following basic operations. (1) For an arbitrary point x, determine if x is in $D(p,q)$, $J(p,q)$ or $D(q,p)$; (2) Given a point x on $J(p,q)$, determine the next vertical tangent point or the next intersection with $J(s,t)$ or a straight line along one direction of $J(p,q)$; (3) For two points x,y on $J(p,q)$, determine the in-front/behind relation along one direction of $J(p,q)$; (4) For two points x and y compare them by x-coordinate, where x and y are intersection points or points of vertical tangency of the bisectors.

3 Randomized Divide and Conquer Algorithm

3.1 Refined Diagram

We first refine $V_k(S)$ and partition it into vertical trapezoids.

Definition 5. *The refined order-k Voronoi diagram $\mathcal{V}_k(S)$ of S is derived by superimposing $V_k(S)$ and $V_{k+1}(S)$. It is defined as:*

$$\mathcal{V}_k(S) = V_k(S) \cup \bigcup_{H\subset S, |H|=k} V_1(S\setminus H)\cap VR_k(H,S).$$

A region $\mathcal{VR}_k(p,H,S)$ of $\mathcal{V}_k(S)$ is associated with a site $p\in S$, which is called the dominator, and a k-element subset $H\subset S$. For any point $x\in\mathcal{VR}_k(p,H,S)$, H is the set of k nearest sites to x and p is the $(k{+}1)$-nearest site to x.

Definition 6. *The* vertical decomposition *of $\mathcal{V}_k(S)$, denoted by $\mathcal{V}_k^{\triangle}(S)$, is the subdivision of the plane into (pseudo-)trapezoids obtained by shooting vertical rays up and down from each vertex in $\mathcal{V}_k(S)$ and each vertical tangent point of each edge in $\mathcal{V}_k(S)$, until the intersection with an edge or all the way to infinity.*

Lemma 2. *$\mathcal{V}_k^{\triangle}(S)$ can be constructed from $V_k(S)$ in expected $O(k(n-k)\log n)$ operations.*

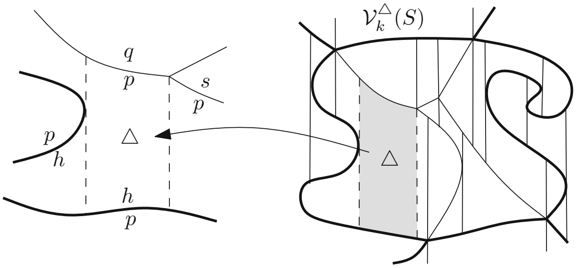

Fig. 1. Trapezoid $\triangle$ of $\mathcal{V}_k^{\triangle}(S)$. $V_k(S)$ is depicted in bold.

A trapezoid $\triangle$ of $\mathcal{V}_k^{\triangle}(S)$ in $\mathcal{VR}_k(p, H, S)$ is defined by the dominator p and 1-4 other sites. Vertical boundaries of the trapezoid may be defined either by an intersection point or by a point of vertical tangency. Moreover, one of the vertical boundaries may be degenerate. Let $d(\triangle)$ be the dominator of the trapezoid and $B(\triangle)$ be the set of sites that together with the dominator define the boundaries of the trapezoid $\triangle$. Then $1 \le |B(\triangle)| \le 4$ and for any point $x \in \triangle$, $H \setminus B(\triangle)$ are the $k - |H \cap B(\triangle)|$ nearest sites to x.

In Fig. 1, the top and bottom edges of $\triangle$ are defined by $J(p, q)$ and $J(p, h)$, respectively, and the left and right edges are defined by a vertical tangent point of $J(p, h)$ and an intersection between $J(p, q)$ and $J(p, s)$, respectively. In other words, $B(\triangle) = \{q, h, s\}$ and $d(\triangle) = p$.

Definition 7. *For a trapezoid $\triangle$ of $\mathcal{V}_k^{\triangle}(S)$, a site $s \notin B(\triangle)$* strongly conflicts *with $\triangle$, if $\overline{\triangle} \subset D(s, d(\triangle))$. A site $s \notin B(\triangle)$* weakly conflicts *with $\triangle$, if $\overline{\triangle} \cap D(s, d(\triangle)) \ne \emptyset$. The set of sites $X \subseteq S$ that* strongly, *resp.* weakly conflict *with $\triangle$ is denoted by $X \wedge_s \triangle$, resp. $X \wedge_w \triangle$.*

In general, the set of *strong conflicts* is different from the set of *weak conflicts*, and $X \wedge_s \triangle \subseteq X \wedge_w \triangle$. In Figure 2, set $S = \{p_1, \ldots, p_7, s_1, \ldots, s_4\}$ is the set of line segments in Euclidean space. $R = \{p_1, \ldots, p_7\}$ is the subset of S and $\triangle$ is the trapezoid of $\mathcal{V}_3^{\triangle}(R)$ in $\mathcal{VR}_3(p_1, \{p_2, p_3, p_4\}, R)$. The dominator $d(\triangle)$ of the trapezoid $\triangle$ is p_1. The set of the sites $B(\triangle)$ that define the boundaries of the trapezoid $\triangle$ is $\{p_2, p_3, p_5, p_6\}$. Since the sites p_2, p_3, p_5, p_6 define the boundary of the trapezoid they cannot conflict with the trapezoid. However, the site p_4 strongly conflicts with $\triangle$, since $\overline{\triangle} \subset D(p_4, p_1)$. Sites that do not belong to R can also conflict with the trapezoid. Here, site s_1 strongly conflicts with $\triangle$, since $\overline{\triangle} \subset D(s_1, p_1)$. However, site s_2 weakly conflicts with $\triangle$, because the dominance region $D(s_2, p_1)$ does not enclose $\overline{\triangle}$, but only intersects $\overline{\triangle}$. Thus, $S \wedge_s \triangle = \{p_4, s_1\}$, $S \wedge_w \triangle = \{p_4, s_1, s_2\}$. In Lemmata 3, 4 we use *weak* and *strong conflicts* for the upper and lower bounds, respectively.

Lemma 3. *Let R be a subset of S and β be a positive integer. Then for any trapezoid $\triangle$ of $\mathcal{V}_\beta^{\triangle}(R)$, $\beta - 4 \le |R \wedge_s \triangle|$ and $|R \wedge_w \triangle| \le \beta$.*

Proof. Let $\triangle$ be in $\mathrm{VR}_\beta(H, R)$. We want to prove that $H \setminus B(\triangle) \subseteq R \wedge_s \triangle$ and $R \wedge_w \triangle \subseteq H$. Since for each point $x \in \triangle$, H are the β nearest sites of x and

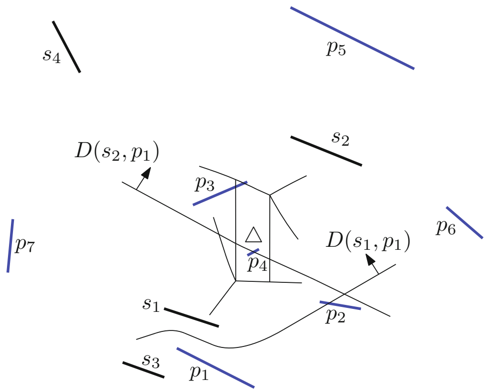

Fig. 2. Trapezoid $\triangle \in \mathcal{VR}_3(p_1, \{p_2, p_3, p_4\})$, where $p_1, \ldots, p_7$ are line segments

$d(\triangle)$ is the $(\beta+1)$-nearest site, for each site $p \in H \setminus B(\triangle)$, $\overline{\triangle} \subset D(p, d(\triangle))$, implying that $H \setminus B(\triangle) \subseteq R \wedge_s \triangle$. For each site $p \in R \wedge_w \triangle$, $D(p, d(\triangle))$ must include $\triangle$; otherwise, $d(\triangle)$ is not the $(\beta+1)$-nearest site for all points in $\triangle$. By Def. 2, p must belong to H, implying that $R \wedge_w \triangle \subseteq H$. □

Lemma 3 and [7, Corollaries 4.3 and 4.4] imply the following.

Lemma 4. *Let R be an r-element random sample of S. Then with probability at least 1/2, as $r \to \infty$, for any $\triangle \in \mathcal{V}_\beta^\triangle(R)$, $|S|/(r-5) \leq |S \wedge_s \triangle|$ and $|S \wedge_w \triangle| \leq \alpha|S|$, where $\beta = O(\log r / \log \log r)$ and $\alpha = O(\log r / r)$.*

Lemma 5. *Let R be a subset of S such that for any trapezoid $\triangle \in \mathcal{V}_\beta^\triangle(R)$, $|S \wedge_s \triangle| > k$. Let v be a Voronoi vertex of $V_k(S)$. Then there exists a trapezoid $\triangle \in \mathcal{V}_\beta^\triangle(R)$ such that v is also a Voronoi vertex of $V_k(S \wedge_w \triangle)$.*

Proof. (Sketch) Let v be a Voronoi vertex incident to Voronoi regions $\mathrm{VR}_k(H_1, S)$, $\mathrm{VR}_k(H_2, S)$ and $\mathrm{VR}_k(H_3, S)$, and let $\triangle$ be a trapezoid of $\mathcal{V}_\beta^\triangle(R)$ such that $v \in \overline{\triangle}$. We want to prove that $H_1 \cup H_2 \cup H_3 \subseteq S \wedge_w \triangle$, which leads to this lemma.

Let H be $H_1 \cup H_2 \cup H_3$ and $t = |H|$. By Definition 1 and Definition 3, t is $k+1$ or $k+2$, and in π_v^S, $s_1 \leq_v \ldots \leq_v s_{t-3} <_v s_{t-2} =_v s_{t-1} =_v s_t <_v s_{t+1} \ldots$, and $H = \{s_1, \ldots, s_t\}$.

Let k' be $|S \wedge_s \triangle|$. By Definition 7, for each site $p \in S \wedge_s \triangle$, $p <_v d(\triangle)$. Therefore, there exists $k'' \geq k'$ such that in π_v^S, $s_{k''-1} <_v s_{k''}$ and either $s_{k''} = d(\triangle)$ or $s_{k''} \leq_v d(\triangle)$, implying that $\{s_1, \ldots, s_{k''-1}\} \subseteq S \wedge_w \triangle$.

Since $k'' > k$ and $t = k+1$ or $k+2$, we have $k'' > t$; otherwise, $k'' = t$ or $t-1$, contradicting either $s_{k''-1} <_v s_{k''}$ or $s_{t-2} =_v s_{t-1} =_v s_t$.

To conclude, $H = \{s_1, \ldots, s_t\} \subseteq \{s_1, \ldots, s_{k''-1}\} \subseteq S \wedge_w \triangle$. Thus v is a Voronoi vertex of $V_k(S \wedge_w \triangle)$. □

3.2 Computing the Voronoi Vertices of $V_k(S)$

Lemma 5 indicates that if for any $\triangle \in \mathcal{V}_\beta^\triangle(R)$, $|S \wedge_s \triangle| > k$, then computing the Voronoi vertices of $V_k(S)$ can be transformed into computing the Voronoi vertices of $V_k(S \wedge_w \triangle)$ for each $\triangle$. Lemma 4 states that on average it takes two trials to generate a sample R such that $|S \wedge_s \triangle| \geq |S|/(r-5)$, where the size r of the random sample R is any sufficiently large constant. Therefore, if $|S|/(r-5) > k$, then we need two trials on average to generate a random sample that satisfies the conditions of Lemma 5. The condition $|S \wedge_w \triangle| \leq \alpha |S|$ in Lemma 4 bounds the depth of the recursion. Following Clarkson [7], the algorithm to compute the Voronoi vertices of $V_k(S)$ is summarized as follows:

- If $|S|/(r-5) \leq k$, compute the vertices of $V_k(S)$ by the algorithm in Section 5.
- Otherwise ($|S|/(r-5) > k$)
  1. Choose $R \subset S$ of size r until R satisfies the conditions of Lemma 4
     (a) Construct $V_\beta(R)$ by the algorithm in Section 4 and Compute $\mathcal{V}_\beta^\triangle(R)$ from $V_\beta(R)$ (Lemma 2).
     (b) Check each trapezoid in $\mathcal{V}_\beta^\triangle(R)$ to satisfy the conditions of Lemma 4.
  2. For each trapezoid $\triangle \in \mathcal{V}_\beta^\triangle(R)$
     (a) Recursively compute the Voronoi vertices of $V_k(S \wedge_w \triangle)$.
     (b) Select vertices of $V_k(S \wedge_w \triangle)$ that are vertices of $V_k(S)$.

3.3 Analysis

Lemma 6. *$V_k(S)$ can be computed from its Voronoi vertices in $O(k(n-k)\log n)$ operations.*

Proof. For points-sites, a vertex is uniquely defined by three sites [14]. Also for point-sites two vertices are adjacent iff their corresponding triples of sites have two sites in common. However, in the abstract setting, three sites may define one or two vertices and the adjacency property does not hold. Therefore, we cannot solve this problem by just using radix sort as it was done for point-sites [7].

Here, in the abstract setting, we use radix sort to extract for each bisector all Voronoi vertices that lie on it, in total $O(|V|)$ operations, where V is the set of vertices in $V_k(S)$. We also assume the existence of a sufficiently large closed curve Γ such that no two bisectors intersect outside Γ.

Consider a set of $m_J > 0$ Voronoi vertices that belong to bisector J (including the artificial Voronoi vertices formed by the intersection between $V_k(S)$ and Γ). m_J must be even; otherwise, at least one Voronoi vertex has no Voronoi edge. We can sort the m_J Voronoi vertices along one direction of J as $v_1, v_2, \ldots, v_{m_J}$ in $O(m_J \log m_J)$ operations, and then link $\overline{v_{2i-1}v_{2i}}$ for $1 \leq i \leq m_J/2$ as Voronoi edges in $O(m_J)$ operations. Therefore, we can compute all the Voronoi edges on J in $O(m_J \log m_J)$ operations. Since $|V|$ is $O(k(n-k))$, the total number of operations is $O(|V|) + \sum_{J \in \mathcal{J}, m_J > 0} O(m_J \log m_J) = O(|V| \log |V|) = O(k(n-k)\log n)$. □

Theorem 1. *$V_k(S)$ can be computed in expected $O(kn^{1+\varepsilon})$ operations, where $\varepsilon > 0$, and the constant factor of the asymptotic bound depends on ε.*

Proof. Recall that r is a sufficiently large constant, $\alpha = O(\log r/r)$ and $\beta = O(\log r/\log\log r)$. There are two cases: (1) If $|S|/(r-5) \le k$, then we use the algorithm from Section 5 to compute the vertices of the order-k Voronoi diagram in expected $O(n^2 2^{\alpha(n)} \log n)$ operations, i.e. $O(r^2k^2\log^2 r\log^2 k)$; (2) If $|S|/(r-5) > k$ then the algorithm proceeds as follows:

1. Choose a random sample that satisfies the conditions of Lemma 4. Do the check by constructing $V_\beta(R)$ and computing $\mathcal{V}_\beta^\triangle(R)$ from $V_\beta(R)$. The construction of $V_\beta(R)$ takes expected $O(r\beta^2\log r)$ operations (see Section 4), and computing $\mathcal{V}_\beta^\triangle(R)$ takes additional expected $O(\beta(r-\beta)\log r)$ operations. The number of the trapezoids in $\mathcal{V}_\beta^\triangle(R)$ is $O(r\beta)$, and the number of operations required to check the sample is $O(nr\beta) \subset O(nr\log r)$.
2. For each trapezoid in $\mathcal{V}_\beta^\triangle(R)$ compute the order-k vertices using recursion. The number of recursive calls is $O(r\beta) \subset O(r\log r)$. Each recursive call inputs $O(\alpha n) = O(n\log r/r)$ sites and outputs $O(\alpha nk)$ vertices. Therefore, the expected total number of operations required to validate each vertex of each recursive call is $O(\alpha nkr\log r)$ which is $O(nk\log^2 r)$.

Therefore, the expected number $t(n)$ of operations for computing the Vononoi vertices of $V_k(S)$ is

$$t(n) \le O\left(r^2k^2\log^2 r\log^2 k\right),\ \ n \le k(r-5)$$
$$t(n) \le O\left(nr\log r\right) + O\left(nk\log^2 r\right) + O(r\log r)t\left(O(n\log r/r)\right),\ \ n > k(r-5),$$

and the depth of the recursion is $O(\log(n/k)/\log(r/\log r))$.

Following [7, Lemma 6.4], if n tends to infinity, $t(n)$ is $O(kn^{1+\varepsilon})$. Since $V_k(S)$ can be constructed from the Voronoi vertices of $V_k(S)$ in expected $O(k(n-k)\log n)$ operations (Lemma 2), $V_k(S)$ can be constructed in expected $O(kn^{1+\varepsilon})$ operations. □

4 First Sub-Algorithm: Iterative Construction

The order-k abstract Voronoi diagram can be computed iteratively similarly to point sites in the Euclidean metric [14]. The following lemma proves the main property used in the iterative construction.

Lemma 7. *Let F be a face of $VR_j(H,S)$ and let $VR_j(H_i,S)$, $1 \le i \le \ell$ be the adjacent regions. Then $V_{j+1}(S) \cap F = V_1(Q) \cap F$, where $Q = \bigcup_{1\le i\le \ell} H_i \setminus H$.*

Proof. We want to show $V_1(Q) \cap F = V_{j+1}(S) \cap F$ which is equal to $V_1(S \setminus H) \cap F$.

Let $x \in \mathrm{VR}_1(s, S\setminus H) \cap F$. For the sake of a contradiction assume $s \notin Q$. This means $s <_x q$, for any $q \in Q$ and thus $x \in \mathrm{VR}_{j+1}(H \cup \{s\})$. Let F' be the face of $\mathrm{VR}_{j+1}(H\cup\{s\})$ that contains x. Since $s \notin Q$, F' does not intersect ∂F, implying that $F' \cap V_j(S)$ is empty. This leads to a contradiction since $F' \cap V_j(S) = F' \cap V_{n-1}(H \cup \{s\})$ and this is nonempty [3, Lemmata 12 and 13]. Hence $V_1(S\setminus H) \cap F = V_1(Q) \cap F$ which finishes the proof. □

Lemma 7 implies that we can compute $V_{j+1}(S)$ by partitioning each face of $V_j(S)$ with the nearest-neighbor Voronoi diagram, which in turn can be computed using the algorithm in [12].

Theorem 2. *$V_k(S)$ can be computed in expected $O(k^2 n \log n)$ operations.*

5 Second Sub-Algorithm: Random Walk Method

We construct $V_k(S)$ by computing $\partial\mathrm{VN}_k(p, S)$ for every $p \in S$, i.e., all the Voronoi edges of $V_k(S)$ belonging to $J(p, q)$. Chazelle and Edelsbrunner [6] computed $\partial\mathrm{VN}_k(p, S)$ based on dynamic convex hulls and the fact that $\mathrm{VN}_k(p, S)$ is simply connected. However, dynamic convex hulls are not applicable in the abstract setting. Since $\mathrm{VN}_k(p, S)$ is simply connected, we can adopt Har-Peled's [9] random walk algorithm to compute $\partial\mathrm{VN}_k(S)$.

$\partial\mathrm{VN}_k(p, S)$ is a substructure of the arrangement of $n-1$ bisectors $\mathcal{J}(p) = \{J(p, q) \mid q \in S \setminus \{p\}\}$, where the bisectors in $\mathcal{J}(p)$ are not x-monotone, but they have constant number of vertical tangency points. Therefore, the structural complexities of the arrangement and its vertical decomposition are of the same asymptotic magnitude. We construct $\partial\mathrm{VN}_k(p, S)$ in the following way: (1) For each connected component of $\partial\mathrm{VN}_k(p, S)$ compute a starting point; (2) For each starting point, traverse the corresponding part of $\partial\mathrm{VN}_k(p, S)$.

Lemma 8 states that starting points can be computed in $O(n \log n)$ expected time. As we walk we can determine the next direction in $O(1)$ time.

Lemma 8. *The starting points of $\partial VN_k(p, S)$ for each of its connected components can be computed in total $O(n \log n)$ expected time.*

Following [9], the expected number of operation required to compute the boundary of the k-neighborhood by the random walk is $O(\lambda_{t+2}(n + m) \log n)$, where t is the maximum number of intersections between two bisectors, and m is the complexity of $\partial\mathrm{VN}_k(p, S)$. In the abstract case, we can show that $t = 2$, i.e. each pair of bisectors $J(p, q)$ and $J(p, r)$ in $\mathcal{J}(p)$ intersect at most twice. Consider $V_1(\{p, q, r\})$. Axiom (A1) implies that each region in this diagram is connected, therefore $V_1(\{p, q, r\})$ has at most two vertices. Thus, $J(p, q)$ and $J(p, r)$ intersect at most twice and $t = 2$.

The main difference between computing the zone in the original version of the algorithm [9] and computing $\partial\mathrm{VN}_k(p, S)$ is that the latter is additionally augmented by the vertical rays from the points of vertical tangency. However, since each bisector allows only a constant number of points of vertical tangency, the expected number of operations increases only by a constant factor.

Theorem 3. *$V_k(S)$ can be computed in expected $O(n^2 2^{\alpha(n)} \log n)$ operations.*

References

1. Agarwal, P.K., de Berg, M., Matoušek, J., Schwarzkopf, O.: Constructing levels in arrangements and higher order Voronoi diagrams. SIAM Journal on Computing **27**(3), 654–667 (1998)

2. Aurenhammer, F., Schwarzkopf, O.: A simple on-line randomized incremental algorithm for computing higher order Voronoi diagrams. International Journal of Computational Geometry and Applications **2**(4), 363–381 (1992)
3. Bohler, C., Cheilaris, P., Klein, R., Liu, C.-H., Papadopoulou, E., Zavershynskyi, M.: On the complexity of higher order abstract Voronoi diagrams. In: Fomin, F.V., Freivalds, R., Kwiatkowska, M., Peleg, D. (eds.) ICALP 2013, Part I. LNCS, vol. 7965, pp. 208–219. Springer, Heidelberg (2013)
4. Boissonnat, J.D., Devillers, O., Teillaud, M.: A semidynamic construction of higher-order Voronoi diagrams and its randomized analysis. Algorithmica **9**, 329–356 (1993)
5. Chan, T.M.: Random sampling, halfspace range reporting, and construction of ($\leq k$)-levels in three dimensions. SIAM Journal on Computing **30**(2), 561–572 (1998)
6. Chazelle, B., Edelsbrunner, H.: An improved algorithm for constructing kth-order Voronoi Diagram. IEEE Transactions on Computers **36**(11), 1349–1454 (1987)
7. Clarkson, K.L.: New applications of random sampling in computational geometry. Discrete and Computational Geometry **2**(1), 195–222 (1987)
8. Gemsa, A., Lee, D.T., Liu, C.-H., Wagner, D.: Higher order city Voronoi diagrams. In: Fomin, F.V., Kaski, P. (eds.) SWAT 2012. LNCS, vol. 7357, pp. 59–70. Springer, Heidelberg (2012)
9. Har-Peled, S.: Taking a walk in a planar arrangment. SIAM Journal on Computing **30**(4), 1341–1367 (2000)
10. Klein, R.: Concrete and Abstract Voronoi Diagrams. LNCS, vol. 400. Springer, Heidelberg (1989)
11. Klein, R., Langetepe, E., Nilforoushan, Z.: Abstract Voronoi Diagrams Revisited. Computational Geometry: Theory and Applications **42**(9), 885–902 (2009)
12. Klein, R., Mehlhorn, K., Meiser, S.: Randomized Incremental Construction of Abstract Voronoi Diagrams. Computational Geometry: Theory and Applications **3**(1), 157–184 (1993)
13. Mehlhorn, K., Meiser, S., Ó'Dúnlaing, C.: On the Construction of Abstract Voronoi Diagrams. Discrete and Computational Geometry **6**(1), 211–224 (1991)
14. Lee, D.T.: On k Nearest Neighbor Voronoi Diagrams in the Plane. IEEE Trans. Computers **31**(6), 478–487 (1982)
15. Liu, C.-H., Lee, D.T.: Higher-order geodesic Voronoi diagrams in a polygonal domain with holes. In: 2013 ACM-SIAM Symposium on Discrete Algorithms, pp. 1633–1645 (2013)
16. Liu, C.-H., Papadopoulou, E., Lee, D.T.: An output-sensitive approach for the L_1/L_∞ k-Nearest-Neighbor Voronoi diagram. In: Demetrescu, C., Halldórsson, M.M. (eds.) ESA 2011. LNCS, vol. 6942, pp. 70–81. Springer, Heidelberg (2011)
17. Papadopoulou, E., Zavershynskyi, M.: On Higher Order Voronoi Diagrams of Line Segments. In: Chao, K.-M., Hsu, T.-s., Lee, D.-T. (eds.) ISAAC 2012. LNCS, vol. 7676, pp. 177–186. Springer, Heidelberg (2012)
18. Ramos, E.: On range reporting, ray shooting, and k-level construction. In: 15th ACM Symposium on Computational Geometry, pp. 390–399 (1999)
19. Zavershynskyi, M., Papadopoulou, E.: A sweepline algorithm for higher order Voronoi diagrams. In: Proc. 10th International Symposium on Voronoi Diagrams in Science and Engineering (ISVD). IEEE-CS (2013)