# Abstract Voronoi-Like Graphs: Extending Delaunay's Theorem and Applications

**Evanthia Papadopoulou**
Faculty of Informatics, Università della Svizzera italiana, Lugano, Switzerland

## Abstract

Any system of bisectors (in the sense of abstract Voronoi diagrams) defines an arrangement of simple curves in the plane. We define *Voronoi-like graphs* on such an arrangement, which are graphs whose vertices are *locally Voronoi*. A vertex $v$ is called locally Voronoi, if $v$ and its incident edges appear in the Voronoi diagram of three sites. In a so-called admissible bisector system, where Voronoi regions are connected and cover the plane, we prove that any Voronoi-like graph is indeed an abstract Voronoi diagram. The result can be seen as an abstract dual version of Delaunay's theorem on (locally) empty circles.

Further, we define Voronoi-like cycles in an admissible bisector system, and show that the Voronoi-like graph induced by such a cycle $C$ is a unique tree (or a forest, if $C$ is unbounded). In the special case where $C$ is the boundary of an abstract Voronoi region, the induced Voronoi-like graph can be computed in expected linear time following the technique of [Junginger and Papadopoulou SOCG'18]. Otherwise, within the same time, the algorithm constructs the Voronoi-like graph of a cycle $C'$ on the same set (or subset) of sites, which may equal $C$ or be enclosed by $C$. Overall, the technique computes abstract Voronoi (or Voronoi-like) trees and forests in linear expected time, given the order of their leaves along a Voronoi-like cycle. We show a direct application in updating a constraint Delaunay triangulation in linear expected time, after the insertion of a new segment constraint, simplifying upon the result of [Shewchuk and Brown CGTA 2015].

**2012 ACM Subject Classification** Theory of computation → Computational geometry

**Keywords and phrases** Voronoi-like graph, abstract Voronoi diagram, Delaunay's theorem, Voronoi tree, linear-time randomized algorithm, constraint Delaunay triangulation

**Digital Object Identifier** 10.4230/LIPIcs.SoCG.2023.52

**Related Version** *Full Version*: https://arxiv.org/abs/2303.06669

**Funding** Supported in part by the Swiss National Science Foundation, project 200021E-201356.

**Acknowledgements** I would like to thank Kolja Junginger for preliminary discussions, numerous figures and comments on earlier versions of this work. I would also like to thank Franz Aurenhammer for valuable suggestions, and Elena Arseneva for constructive comments and figures.

## 1 Introduction

Delaunay's theorem [6] is a well-known cornerstone in Computational Geometry: given a set of points, a triangulation is *globally Delaunay* if and only if it is *locally Delaunay*. A triangulation edge is called *locally Delaunay* if it is incident to only one triangle, or it is incident to two triangles, and appears in the Delaunay triangulation of the four related vertices. The Voronoi diagram and the Delaunay triangulation of a point set are dual to each other. These two highly influential and versatile structures are often used and computed interchangeably; see the book of Aurenhammer et al. [2] for extensive information.

Let us pose the following question: how does Delaunay's theorem extend to Voronoi diagrams of generalized (not necessarily point) sites? We are interested in simple geometric objects in the plane such as line segments, polygons, disks, or point clusters, as they often appear in application areas, and answering this question is intimately related to efficient construction algorithms for Voronoi diagrams (or their duals) on these objects. We consider this question in the framework of abstract Voronoi diagrams in the plane [11] so that we can simultaneously answer it for various concrete and fundamental cases under their umbrella.

Although Voronoi diagrams and Delaunay triangulations of point sites have been widely used in many fields of science, being available in most software libraries of commonly used programming languages, practice has not been the same for their counterparts of simple geometric objects. In fact it is surprising that certain related questions may have remained open or non-optimally solved. Edelsbrunner and Seidel [7] defined Voronoi diagrams as lower envelopes of distance functions in a space one dimension higher, making a powerful link to arrangements, which made their rich combinatorial and algorithmic results applicable, e.g., [15]. However, there are different levels of difficulty concerning arrangements of planes versus more general surfaces, which play a role, especially in practice.

In this paper we define Voronoi-like graphs based on local information, inspired by Delaunay's theorem. Following the framework of abstract Voronoi diagrams (AVDs) [11], let $S$ be a set of $n$ abstract sites (a set of indices) and $\mathcal{J}$ be their underlying system of bisectors, which satisfies some simple combinatorial properties (see Sections 2, 3). Consider a graph $G$ on the arrangement of the bisector system possibly truncated within a simply connected domain $D$. The vertices of $G$ are vertices of the bisector arrangement, its leaves lie on the boundary $\partial D$, and the edges are maximal bisector arcs connecting pairs of vertices. A vertex $v$ in $G$ is called *locally Voronoi*, if $v$ and its incident edges within a small neighborhood around $v$ appear in the Voronoi diagram of the three sites defining $v$ (Def. 3), see Figure 4. The graph $G$ is called *Voronoi-like*, if its vertices (other than its leaves on $\partial D$) are locally Voronoi vertices (Def. 4), see Figure 5. If the graph $G$ is a simple cycle on the arrangement of bisectors related to one site $p$ and its vertices are locally Voronoi of degree 2, then it is called a *Voronoi-like cycle*, for brevity a *site-cycle* (Def. 10).

A major difference between points in the Euclidean plane, versus non-points, such as line segments, disks, or AVDs, can be immediately pointed out: in the former case the bisector system is a line arrangement, while in the latter, the bisecting curves are not even pseudolines. On a line arrangement, it is not hard to see that a Voronoi-like graph coincides with the Voronoi diagram of the involved sites: any Voronoi-like cycle is a convex polygon, which is, in fact, a Voronoi region in the Voronoi diagram of the relevant sites. But in the arrangement of an abstract bisector system, many different Voronoi-like cycles can exist for the same set of sites, see, e.g., Figure 10. Whether a Voronoi-like graph corresponds to a Voronoi diagram is not immediately clear.

In this paper we show that a Voronoi-like graph on the arrangement of an abstract bisector system is as close as possible to being an abstract Voronoi diagram, subject to, perhaps, *missing* some faces (see Def. 5). If the graph misses no face, then it is a Voronoi diagram. Thus, in the classic AVD model [11], where abstract Voronoi regions are connected and cover the plane, any Voronoi-like graph is indeed an abstract Voronoi diagram. This result can be seen as an abstract dual version of Delaunay's theorem.

Voronoi-like graphs (and their duals) can be very useful structures to hold partial Voronoi information, either when dealing with disconnected Voronoi regions, or when considering partial information concerning some region. Building a Voronoi-like graph of partial information may be far easier than constructing the full diagram. In some cases, the full diagram may even be undesirable as in the example of Section 6 in updating a constrained Delaunay triangulation.

The term *Voronoi-like diagram* was first used, in a restricted sence, by Junginger and Papadopoulou [8], defining it as a tree (occasionally a forest) that subdivided a planar region enclosed by a so-called *boundary curve* defined on a subset of Voronoi edges. Their Voronoi-like diagram was then used as an intermediate structure to perform deletion in an abstract Voronoi diagram in linear expected time. In this paper the formulation of a Voronoi-like graph is entirely different; we nevertheless prove that the Voronoi-like diagram of [8] remains a special case of the one defined in this paper. We thus use the results of [8] when applicable, and extend them to Voronoi-like cycles in an admissible bisector system.

In the remainder of this section we consider an *admissible* bisector system $\mathcal{J}$ following the classic AVD model [11], where bisectors are unbounded simple curves and Voronoi regions are connected. To avoid issues with infinity, we asume a large Jordan curve $\Gamma$ (e.g, a circle) bounding the computation domain, which is large enough to enclose any bisector intersection. In the sequel, we list further results, which are obtained in this paper under this model.

We consider a Voronoi-like cycle $C$ on the arrangement of bisectors $\mathcal{J}_p \subseteq \mathcal{J} \cup \Gamma$, which are *related* to a site $p \in S$. Let $S_C \subseteq S \setminus \{p\}$ be the set of sites that (together with $p$) contribute to the bisector arcs in $C$. The cycle $C$ encodes a *sequence of site occurrences* from $S_C$. We define the Voronoi-like graph $\mathcal{V}_l(C)$, which can be thought of as a Voronoi diagram of *site occurrences*, instead of sites, whose order is represented by $C$. We prove that $\mathcal{V}_l(C)$ is a tree, or a forest if $C$ is unbounded, and it exists for any Voronoi-like cycle $C$. The uniqueness of $\mathcal{V}_l(C)$ can be inferred from the results in [8]. The same properties can be extended to Voronoi-like graphs of cycles related to a set $P$ of $k$ sites.

We then consider the randomized incremental construction of [8], and apply it to a Voronoi-like cycle in linear expected time. If $C$ is the boundary of a Voronoi region then $\mathcal{V}_l(C)$, which is the part of the abstract Voronoi diagram $\mathcal{V}(S_C)$, truncated by $C$, can be computed in expected linear time (this was previously shown [8, 10]). Otherwise, within the same time, the Voronoi-like graph of a (possibly different) Voronoi-like cycle $C'$, enclosed by $C$, is computed by essentially the same algorithm. We give conditions under which we can force the randomized algorithm to compute $\mathcal{V}_l(C)$, if desirable, without hurting its expected-linear time complexity, using deletion [8] as a subroutine. The overall technique follows the randomized linear-time paradigm of Chew [5], originally given to compute the Voronoi diagram of points in convex position. The generalization of Chew's technique can potentially be used to convert algorithms working on point sites, which use it, to counterparts involving non-point sites that fall under the umbrella of abstract Voronoi diagrams.

Finally, we give a direct application for computing the Voronoi-like graph of a site-cycle in linear expected time, when updating a constrained Delaunay triangulation upon insertion of a new line segment, simplifying upon the corresponding result of Shewchuk and Brown[16]. The resulting algorithm is extremely simple. By modeling the problem as computing the dual of a Voronoi-like graph, given a Voronoi-like cycle (which is not a Voronoi region's boundary), the algorithmic description becomes almost trivial and explains the technicalities, such as self-intersecting subpolygons, that are listed by Shewchuk and Brown.

The overall technique computes abstract Voronoi, or Voronoi-like, trees and forests in linear expected time, given the order of their leaves along a Voronoi-like cycle. In an extended paper, we also give simple conditions under which the cycle $C$ is an arbitrary Jordan curve of constant complexity. All omitted proofs appear in [14].

## 2 Preliminaries and definitions

We follow the framework of abstract Voronoi diagrams (AVDs), which have been defined by Klein [11]. Let $S$ be a set of $n$ abstract sites (a set of indices) and $\mathcal{J}$ be an underlying system of bisectors that satisfy some simple combinatorial properties (some axioms). The

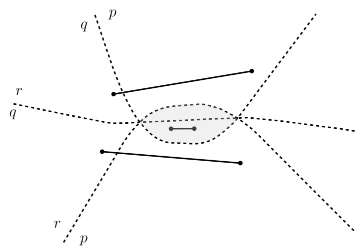


**Figure 1** Related segment bisectors intersecting twice. $\mathrm{VR}(p, \{p, q, r\})$ is shaded.

bisector $J(p, q)$ of two sites $p, q \in S$ is a simple curve that subdivides the plane into two open domains: the *dominance region of* $p$, $D(p, q)$, having label $p$, and the *dominance region of* $q$, $D(q, p)$, having label $q$.

The *Voronoi region* of site $p$ is

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

The *Voronoi diagram* of $S$ is $\mathcal{V}(S) = \mathbb{R}^2 \setminus \bigcup_{p \in S} \mathrm{VR}(p, S)$. The vertices and the edges of $\mathcal{V}(S)$ are called *Voronoi vertices* and *Voronoi edges*, respectively.

Variants of abstract Voronoi diagrams of different degrees of generalization have been proposed, see e.g., [12, 3]. Following the original formulation by Klein [11], the bisector system $\mathcal{J}$ is called *admissible*, if it satisfies the following axioms, for every subset $S' \subseteq S$:

**(A1)** Each Voronoi region $\mathrm{VR}(p, S')$ is non-empty and pathwise connected.
**(A2)** Each point in the plane belongs to the closure of a Voronoi region $\mathrm{VR}(p, S')$.
**(A3)** Each bisector is an unbounded simple curve homeomorphic to a line.
**(A4)** Any two bisectors intersect transversally and in a finite number of points.

Under these axioms, the abstract Voronoi diagram $\mathcal{V}(S)$ is a planar graph of complexity $O(n)$, which can be computed in $O(n \log n)$ time, randomized [13] or deterministic [11].

To avoid dealing with infinity, we assume that $\mathcal{V}(S)$ is truncated within a domain $D_\Gamma$ enclosed by a large Jordan curve $\Gamma$ (e.g., a circle or a rectangle) such that all bisector intersections are contained in $D_\Gamma$. Each bisector crosses $\Gamma$ exactly twice and transversally. All Voronoi regions are assumed to be truncated by $\Gamma$, and thus, lie within the domain $D_\Gamma$.

We make a general position assumption that no three bisectors involving one common site intersect at the same point, that is, all vertices in the arrangement of the bisector system $\mathcal{J}$ have degree 6, and Voronoi vertices have degree 3.

Bisectors that have a site $p$ in common are called *related*, in particular, *$p$-related*. Let $\mathcal{J}_p \subseteq \mathcal{J}$ denote the set of all $p$-related bisectors in $\mathcal{J}$. Under axiom A2, if related bisectors $J(p, q)$ and $J(p, s)$ intersect at a vertex $v$, then $J(q, s)$ must also intersect with them at the same vertex, which is a Voronoi vertex in $V(\{p, q, s\})$ (otherwise, axiom A2 would be violated in $V(\{p, q, s\})$). In an admissible bisector system, related bisectors can intersect at most twice [11]; thus, a Voronoi diagram of three sites may have at most two Voronoi vertices, see e.g., the bisectors of three line segments in Figure 1. The curve $\Gamma$ can be interpreted as a $p$-related bisector $J(p, s_\infty)$, for a site $s_\infty$ representing infinity, for any $p \in S$.

▶ **Observation 1.** *In an admissible bisector system, related bisectors that do not intersect or intersect twice must follow the patterns illustrated in Figures 2 and 3 respectively.*

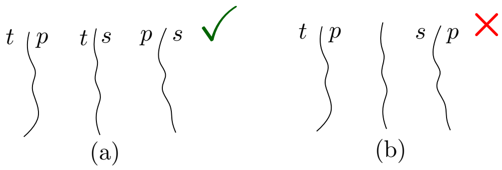



**Figure 2** Non-intersecting bisectors; (a) is legal (✓); (b) is illegal (×).

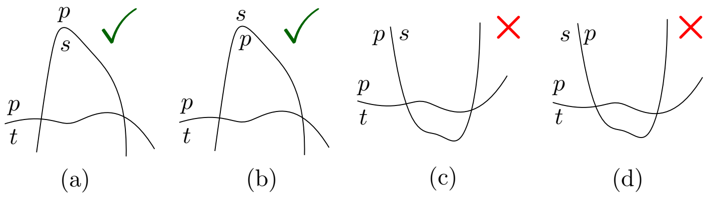



**Figure 3** Bisectors intersecting twice; legal (✓) and illegal(×).

**Proof.** In Figure 3(c) the pattern is illegal because of axiom A1, and in Figure 3(d) because of combining axioms A2 and A1: $J(s,t)$ must pass through the intersection points of $J(p,s)$ and $J(t,p)$, by A2. Then any possible configuration of $J(s,t)$ results in violating either axiom A1 or A2. If bisectors do not intersect, any pattern other than the one in Figure 2(a) can be shown illegal by combining axioms A1 and A2. ◀

▶ **Observation 2** ([8]). *In an admissible bisector system, no cycle in the arrangment of bisectors related to $p$ can have the label $p$ on the exterior of the cycle, for all of its arcs.*

Any component $\alpha$ of a bisector curve $J(p,q)$ is called an *arc*. We use $s_\alpha \in S$ to denote the site such that arc $\alpha \subseteq J(p,s_\alpha)$. Any component of $\Gamma$ is called a $\Gamma$-arc. The arrangement of a bisector set $\mathcal{J}_x \subseteq \mathcal{J}$ is denoted by $\mathcal{A}(\mathcal{J}_x)$.

## 3 Defining abstract Voronoi-like graphs and cycles

In order to define Voronoi-like graphs in a broader sense, we can relax axioms A1-A4 in this section. In particular, we drop axiom A1 to allow disconnected Voronoi regions and relax axiom A3 to allow disconnected (or even closed) bisecting curves. The bisector $J(p,q)$ of two sites $p,q \in S$ still subdivides the plane into two open domains: the *dominance region of* $p$, $D(p,q)$, and the *dominance region of* $q$, $D(q,p)$, however, $D(p,q)$ may be disconnected or bounded. Axioms A2 and A4 remain. Unless otherwise specified, we use the general term *abstract bisector system* to denote such a relaxed variant in the subsequent definitions and in Theorem 6. The term *admissible bisector system* always implies axioms A1-A4.

Let $G=(V,E)$ be a graph on the arrangement of an abstract bisector system $\mathcal{J}$, truncated within a simply connected domain $D \subseteq D_\Gamma$ (the leaves of $G$ are on $\partial D$). The vertices of $G$ are arrangement vertices and the edges are maximal bisector arcs connecting pairs of vertices. Figure 5 illustrates examples of such graphs on a bisector arrangment (shown in grey). Under the general position assumption, the vertices of $G$, except the leaves on $\partial D$, are of degree 3.

▶ **Definition 3.** *A vertex $v$ in graph $G$ is called* locally Voronoi*, if $v$ and its incident graph edges, within a small neighborhood around $v$, $N(v)$, appear in the Voronoi diagram of the set of three sites defining $v$, denoted $S_v$, see Figure 4(a).*

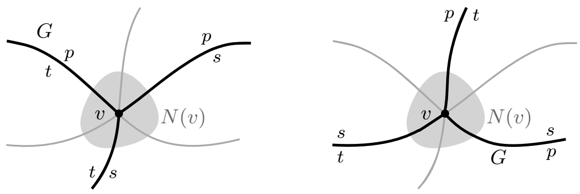
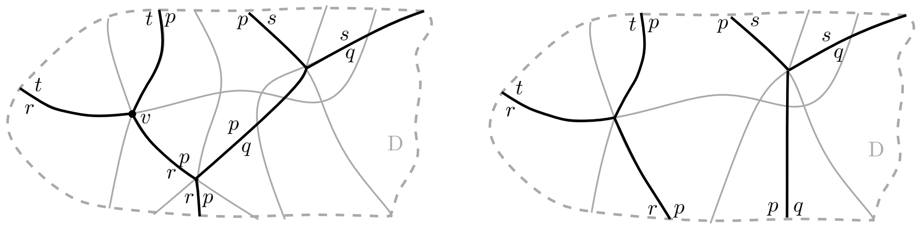



**Figure 4** (a) Vertex $v$ is locally Voronoi: $G \cap N(v) = \mathcal{V}(\{p, s, t\}) \cap N(v)$; $N(v)$ is shaded, $G$ is bold, and bisectors are grey. (b) Vertex $v$ is locally farthest Voronoi.




(a) (b)

**Figure 5** Voronoi-like graphs shown in bold on an arrangement of bisectors shown in grey.

*If instead we consider the farthest Voronoi diagram of $S_v$, then $v$ is called locally Voronoi of the* farthest-type*, see Figure 4(b). An ordinary locally Voronoi vertex is of the* nearest-type*.*

▶ **Definition 4.** *A graph $G$ on the arrangement of an abstract bisector system, enclosed within a simply connected domain $D$, is called* Voronoi-like*, if its vertices (other than its leaves on $\partial D$) are locally Voronoi vertices. If $G$ is disconnected, we further require that consecutive leaves on $\partial D$ have* consistent labels*, i.e., they are incident to the dominance region of the same site, as implied by the incident bisector edges in $G$, see Figure 5.*

The graph $G$ is actually called an *abstract Voronoi-like graph* but, for brevity, we usually skip the term *abstract*. We next consider the relation between a Voronoi-like graph $G$ and the Voronoi diagram $\mathcal{V}(S) \cap D$, where $S$ is the set of sites involved in the edges of $G$. Since the vertices of $G$ are locally Voronoi, each face $f$ in $G$ must have the label of exactly one site $s_f$ in its interior, which is called the *site of $f$*.

▶ **Definition 5.** *Imagine we superimpose $G$ and $\mathcal{V}(S) \cap D$. A face $f$ of $\mathcal{V}(S) \cap D$ is said to be* missing *from $G$, if $f$ is covered by faces of $G$ that belong to sites that are different from the site of $f$, see Figure 7, which is derived from Figure 6.*

▶ **Theorem 6.** *Let $r$ be a face of an abstract Voronoi-like graph $G$ and let $s_r$ denote its site (the bisectors bounding $r$ have the label $s_r$ inside $r$). Then one of the following holds:*

1. *there is a Voronoi face $r'$ in $\mathcal{V}(S) \cap D$, of the same site as $r$, $r' \subseteq VR(s_r, S)$, such that $r' \subseteq r$, see Figure 7.*
2. *face $r$ is disjoint from the Voronoi region $VR(s_r, S)$. Further, it is entirely covered by Voronoi faces of $\mathcal{V}(S) \cap D$, which are* missing *from $G$, see Figure 8.*

**Proof.** Imagine we superimpose $G$ and $\mathcal{V}(S) \cap D$. Face $r$ in $G$ cannot partially overlap any face of the Voronoi region $\mathrm{VR}(s_r, S)$ because if it did, some $s_r$-related bisector, which contributes to the boundary of $r$, would intersect the interior of $\mathrm{VR}(s_r, S)$, which is not

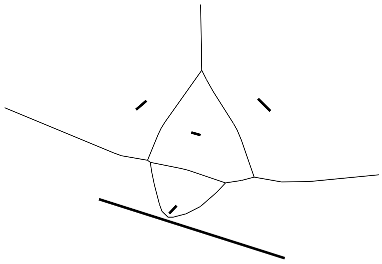


**Figure 6** A Voronoi diagram of 4 segments.

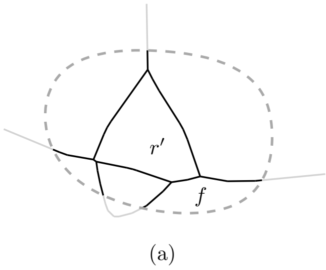



**Figure 7** (a) Voronoi diagram $V(S) \cap D$; (b) Voronoi-like graph $G$; $r' \subseteq r$; $f$ is missing from (b).

possible by the definition of a Voronoi region. For the same reason, $r$ cannot be contained in $\mathrm{VR}(s_r, S)$. Since Voronoi regions cover the plane, the claim, except from the last sentence in item 2, follows.

Consider a Voronoi face $c'$ of $\mathcal{V}(S) \cap D$ that overlaps with face $r$ of $G$ in case 2, where the site of $c'$, $s_c$, is different from $s_r$. Since $c'$ overlaps with $r$, it follows that $c'$ cannot be entirely contained in any face of site $s_c$ in $G$. Furthermore, $c'$ cannot overlap partially with any face of $s_c$ in $G$, by the proof in the previous paragraph. Thus, $c'$ is disjoint from any face of $G$ of site $s_c$, i.e., it must be missing from $G$. In Figure 8, face $c'$ contains $r$. ◀

▶ **Corollary 7.** *If no Voronoi face of $\mathcal{V}(S) \cap D$ is missing from $G$, then $G = \mathcal{V}(S) \cap D$.*

Let us now consider an admissible bisector system, satisfying axioms A1-A4.

▶ **Corollary 8.** *In an* admissible bisector system *$\mathcal{J}$, if $D$ corresponds to the entire plane, then any Voronoi-like graph on $\mathcal{J}$ equals the Voronoi diagram of the relevant set of sites.*

In an admissible bisector system, Voronoi regions are connected, thus, only faces incident to $\partial D$ may be missing from $\mathcal{V}(S) \cap D$.

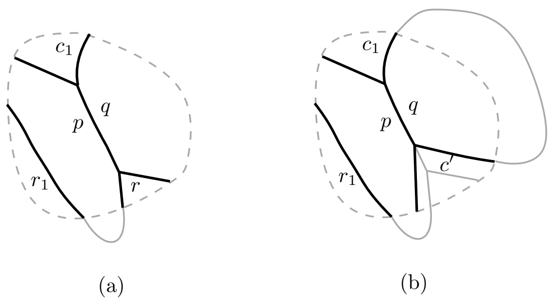



**Figure 8** (a) Voronoi-like graph $G$; (b) $V(S) \cap D$; face $c'$, which covers $r$, is missing from $G$.

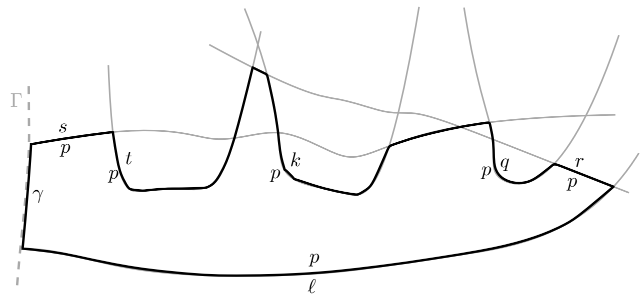



**Figure 9** A Voronoi-like cycle for site $p$. $S_c = \{s, t, k, q, r, l\}$.

▶ **Corollary 9.** *In an admissible bisector system, any face $f$ of $G$ that does not touch $\partial D$ either coincides with or contains the Voronoi region $VR(s_f, S)$.*

Thus, in an admissible bisector system, we need to characterize the faces of a Voronoi-like graph that interact with the boundary of the domain $D$. That is, we are interested in Voronoi-like trees and forests.

Let $p$ be a site in $S$ and let $\mathcal{J}_p$ denote the set of $p$-related bisectors in $\mathcal{J}$.

▶ **Definition 10.** *Let $C$ be a cycle in the arrangement of $p$-related bisectors $\mathcal{A}(\mathcal{J}_p \cup \Gamma)$ such that the label $p$ appears in the interior of $C$. A vertex $v$ in $C$ is called* degree-2 locally Voronoi*, if its two incident bisector arcs correspond to edges in the Voronoi diagram $\mathcal{V}(S_v)$ of the three sites that define $v$ ($p \in S_v$). In particular, $C \cap N(v) \subseteq \mathcal{V}(S_v) \cap N(v)$, where $N(v)$ is a small neighborhood around $v$. The cycle $C$ is called* Voronoi-like*, if its vertices are either degree-2 locally Voronoi or points on $\Gamma$. For brevity, $C$ is also called a $p$-*cycle*, or* site-cycle*, if the site $p$ is not specified. If $C$ bounds a Voronoi region, then it is called a* Voronoi cycle*.*

*$C$ is called* bounded *if it contains no $\Gamma$-arcs, otherwise, it is called* unbounded*.*

The part of the plane enclosed by $C$ is called the *domain of* $C$, denoted as $D_C$. Any $\Gamma$-arc of $C$ indicates an opening of the domain to infinity. Figure 9 illustrates a Voronoi-like cycle for site $p$, which is unbounded (see the $\Gamma$-arc $\gamma$). It is easy to see in this figure that other $p$-cycles exist, on the same set of sites, which may enclose or be enclosed by $C$. The innermost such cycle is the boundary of a Voronoi region, see Figure 10.

Let $S_C \subseteq S \setminus \{p\}$ denote the set of sites that (together with $p$) contribute the bisector arcs of $C$, $S_C = \{s_\alpha \in S \setminus \{p\} \mid \alpha \in C \setminus \Gamma\}$. We refer to $S_C$ as the *set of sites relevant* to $C$. Let $\hat{C}$ denote the Voronoi cycle $\hat{C} = \partial(\mathrm{VR}(p, S_C \cup \{p\}) \cap D_\Gamma)$.

▶ **Observation 11.** *In an admissible bisector system, there can be many different Voronoi-like cycles involving the same set of sites. Any such cycle $C$ must enclose the Voronoi cycle $\hat{C}$. Further, $S_{\hat{C}} \subseteq S_C$. In the special case of a line arrangement, e.g., bisectors of point-sites in the Euclidean plane, a site-cycle $C$ is unique for $S_C$; in particular, $C = \hat{C}$.*

A Voronoi-like cycle $C$ must share several bisector arcs with its Voronoi cycle $\hat{C}$, at least one bisector arc for each site in $S_{\hat{C}}$. Let $C \cap \hat{C}$ denote the sequence of common arcs between $C$ and $\hat{C}$. Several other $p$-cycles $C'$, where $S_{\hat{C}} \subseteq S_{C'} \subseteq S_C$, may lie between $C$ and $\hat{C}$, all sharing Other $p$-cycles may enclose $C$. Figure 10 shows such cycles, where the innermost one is $\hat{C}$; its domain (a Voronoi region) is shown in solid grey.

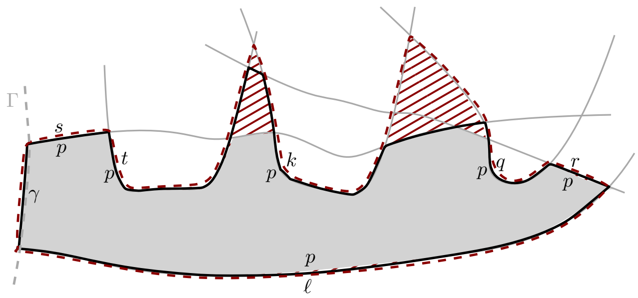



**Figure 10** Voronoi-like cycles for site $p$, $S_c = \{s, t, k, q, r, l\}$.

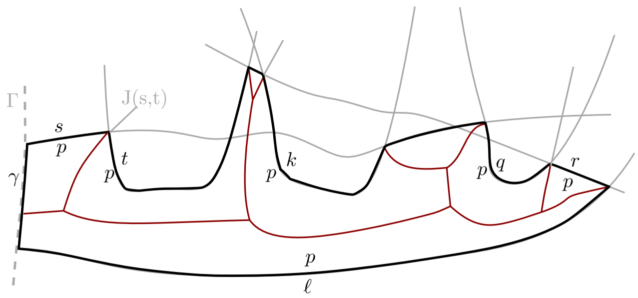



**Figure 11** The Voronoi-like graph $\mathcal{V}_l(C)$ (red tree) of the site-cycle $C$ of Fig. 9.

## 4 The Voronoi-like graph of a cycle

Let $\mathcal{J}$ be an admissible bisector system and let $C$ be a Voronoi-like cycle for site $p$, which involves a set of sites $S_C$ ($p \notin S_C$). Let $\mathcal{J}_C \subseteq \mathcal{J}$ be the subset of all bisectors that are related to the sites in $S_c$. The cycle $C$ corresponds to a sequence of *site-occurrences* from $S_C$, which imply a Voronoi-like graph $\mathcal{V}_l(C)$ in the domain of $C$, defined as follows:

▶ **Definition 12.** *The Voronoi-like graph $\mathcal{V}_l(C)$, implied by a Voronoi-like cycle $C$, is a graph on the underlying arrangement of bisectors $\mathcal{A}(\mathcal{J}_C) \cap D_C$, whose leaves are the vertices of $C$, and its remaining (non-leaf) vertices are locally Voronoi vertices, see Figure 11. (The existence of such a graph on $\mathcal{A}(\mathcal{J}_C) \cap D_C$ remains to be established).*

In this section we prove the following theorem for any Voronoi-like cycle $C$ on $\mathcal{A}(\mathcal{J}_p \cup \Gamma)$.

▶ **Theorem 13.** *The Voronoi-like graph $\mathcal{V}_l(C)$ of a $p$-cycle $C$ has the following properties:*
1. *it exists and is unique;*
2. *it is a tree if $C$ is bounded, and a forest if $C$ is unbounded;*
3. *it can be computed in expected linear time, if it is the boundary of a Voronoi region; otherwise, in expected linear time we can compute $\mathcal{V}_l(C')$ for some $p$-cycle $C'$ that is enclosed by $C$ (possibly, $C' = C$ or $C' = \hat{C}$).*

Recall that $\hat{C}$ denotes the Voronoi-cycle enclosed by $C$, where $\hat{C} = \partial[\mathrm{VR}(p, S_C \cup \{p\}) \cap D_\Gamma]$. Then $\mathcal{V}_l(\hat{C})$ is the Voronoi diagram $\mathcal{V}(S_C) \cap D_{\hat{C}}$. To derive Theorem 13 we show each item separately in subsequent lemmas.

▶ **Lemma 14.** *Assuming that it exists, $\mathcal{V}_l(C)$ is a forest, and if $C$ is bounded, then $\mathcal{V}_l(C)$ is a tree. Each face of $\mathcal{V}_l(C)$ is incident to exactly one bisector arc $\alpha$ of $C$, which is called the face (or region) of $\alpha$, denoted $R(\alpha, C)$.*

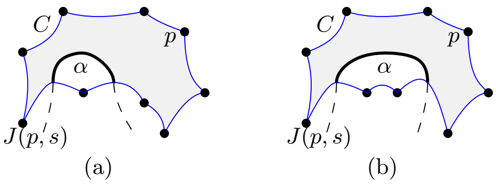



**Figure 12** Three cases of the arc insertion operation.

If $C$ is the boundary of a Voronoi region, the tree property of the Voronoi diagram $\mathcal{V}(S) \cap D_C$ had been previously shown in [8, 4]. Lemma 14 generalizes it to Voronoi-like graphs for any Voronoi-like cycle $C$.

In [8], a *Voronoi-like diagram* was defined as a tree structure subdividing the domain of a so-called *boundary curve*, which was implied by a set of Voronoi edges. A boundary curve is a Voronoi-like cycle but not necessarily vice versa. That is, the tree structure of [8] was defined using some of the properties in Lemma 14 as its definition, and the question whether such a tree always existed had remained open. In this paper a Voronoi-like graph is defined entirely differently, but Lemma 14 implies that the two structures are equivalent within the domain of a boundary curve. As a result, we can use and extend the results of [8].

Given a $p$-cycle $C$, and a bisector $J(p, s)$ that intersects it, an *arc-insertion operation* can be defined as follows [8]. Let $\alpha \subseteq J(p, s)$ be a maximal component of $J(p, s)$ in the domain of $C$, see Figure 12. Let $C_\alpha = C \oplus \alpha$ denote the $p$-cycle obtained by substituting with $\alpha$ the superflous portion of $C$ between the endpoints of $\alpha$. (Note that only one portion of $C$ forms a $p$-cycle with $\alpha$, thus, no ambiguity exists). There are three different main cases possible as a result, see Figure 12: 1) $\alpha$ may lie between two consecutive arcs of $C$, in which case $|C_\alpha| = |C| + 1$; 2) $\alpha$ may cause the deletion of one or more arcs in $C$, thus, $|C_\alpha| \leq |C|$; 3) the endpoints of $\alpha$ may lie on the same arc $\omega$ of $C$, in which case $\omega$ splits in two different arcs, thus, $|C_\alpha| = |C| + 2$. In all cases $C_\alpha$ is enclosed by $C$ ($|\cdot|$ denotes cardinality).

The arc-insertion operation can be naturally extended to the Voronoi-like graph $\mathcal{V}_l(C)$ to insert arc $\alpha$ and obtain $\mathcal{V}_l(C_\alpha)$. We use the following lemma, which can be extracted from [8] (using Theorem 18, Theorem 20, and Lemma 21 of [8]).

▶ **Lemma 15** ([8]). *Given $\mathcal{V}_l(C)$, arc $\alpha \in J(p, s) \cap D_C$, and the endpoints of $\alpha$ on $C$, we can compute the merge curve $J(\alpha) = \partial R(\alpha, C_\alpha)$, using standard techniques as in ordinary Voronoi diagrams. If the endpoints of $\alpha$ lie on different arcs of $C$, or $\Gamma$, the time complexity is $O(|J(\alpha)| + |C \setminus C_\alpha|)$. Otherwise, $\alpha$* splits *a bisector arc $\omega$, and its region $R(\omega, C)$, into $R(\omega_1, C_\alpha)$ and $R(\omega_2, C_\alpha)$; the time complexity increases to $O(|J(\alpha)| + \min\{|R(\omega_1, C_\alpha)|, |R(\omega_2, C_\alpha)|\})$.*

The correctness proofs from [8, 10], which are related to Lemma 15, remain intact if performed on a Voronoi-like cycle, as long as the arc $\alpha$ is contained in the cycle's domain; see also [10, Lemma 9]. Thus, Lemma 15 can be established.

Next we prove the existence of $\mathcal{V}_l(C)$ by construction. To this goal we use a *split relation* between bisectors in $\mathcal{J}_p$ or sites in $S_C$, which had also been considered in [9], see Figure 13.

▶ **Definition 16.** *For any two sites $s, t \in S_C$, we say that $J(p, s)$* splits *$J(p, t)$ (we also say that $s$ splits $t$, with respect to $p$), if $J(p, t) \cap D(p, s)$ contains two connected components.*

From the fact that related bisectors in an admissible bisector system intersect at most twice, as shown in Figs. 2 and 3, we can infer that the split relation is asymmetric and transitive, thus, it is also acyclic. The split relation induces a strict partial order on $S_C$,

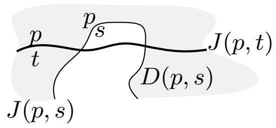



**Figure 13** $J(p,s)$ splits $J(p,t)$. In the partial order, $s <_p t$.

where $s <_p t$, if $J(p,s)$ *splits* $J(p,t)$, see Figure 13. Let $o_p$ be a topological order of the resulting directed acyclic graph, which underlies the split relation on $S_c$ induced by $p$.

The following lemma shows that $\mathcal{V}_l(C)$ exists by construction, see [14]. It builds upon a more-restricted version regarding a boundary curve, which was considered in [9].

▶ **Lemma 17.** *Given the topological ordering of the split relation $o_p$, $\mathcal{V}_l(C)$ can be constructed in $O(|C|^2)$ time; thus, $\mathcal{V}_l(C)$ exists. Further, at the same time, we can construct $\mathcal{V}_l(C')$ for any other Voronoi-like cycle $C'$ that is enclosed by $C$, $S_{C'} \subseteq S_C$.*

The following lemma can also be extracted from [8, 10]. It can be used to establish the uniqueness of $\mathcal{V}_l(C)$. Similarly to Lemma 15, its original statement does not refer to a $p$-cycle, however, nothing in its proof prevents its adaptation to a $p$-cycle, see [10, Lemma 29].

▶ **Lemma 18.** *[10] Let $C$ be a $p$-cycle and let $\alpha, \beta$ be two bisector arcs in $C$, where $s_\alpha \neq s_\beta$. Suppose that a component $e$ of $J(s_\alpha, s_\beta)$ intersects $R(\alpha, C)$. Then $J(p, s_\beta)$ must intersect $D_C$ with a component $\beta' \subseteq J(p, s_\beta) \cap D_C$ such that $e$ is a portion of $\partial R(\beta', C \oplus \beta')$.*

By Lemma 18, if $J(s_\alpha, s_\beta)$ intersects $R(\alpha, C)$, then a face of $s_\beta$ must be *missing* from $\mathcal{V}_l(C)$ (compared to $\mathcal{V}_l(\hat{C})$) implying that an arc of $J(p, s_\beta)$ must be missing from $C$. Thus, $\mathcal{V}_l(C)$ must be unique.

We now use the randomized incremental construction of [8] to construct $\mathcal{V}_l(C)$, which in turn follows Chew [5], to establish the last claim of Theorem 13. Let $o = (\alpha_1, \ldots \alpha_n)$ be a random permutation of the bisector arcs of $C$, where each arc represents a different occurrence of a site in $S_C$. The incremental algorithm works in two phases. In phase 1, delete arcs from $C$ in the reverse order $o^{-1}$, while registering their neighbors at the time of deletion. In phase 2, insert the arcs, following $o$, using their neighbors information from phase 1.

Let $C_i$ denote the $p$-cycle constructed by considering the first $i$ arcs in $o$ in this order. $C_1$ is the $p$-cycle consisting of $J(s_{\alpha_1}, p)$ and the relevant $\Gamma$-arc. Given $C_i$, let $\alpha'_{i+1}$ denote the bisector component of $J(p, s_{\alpha_{i+1}}) \cap D_{C_i}$ that contains $\alpha_{i+1}$ (if any), see Figure 12 where $\alpha$ stands for $\alpha'_{i+1}$. If $\alpha_{i+1}$ lies outside $C_i$, then $\alpha'_{i+1} = \emptyset$ (this is only possible if $C_i$ is not a Voronoi cycle). Let cycle $C_{i+1} = C_i \oplus \alpha'_{i+1}$ (if $\alpha'_{i+1} = \emptyset$, then $C_{i+1} = C_i$). Given $\alpha'_{i+1}$, and $\mathcal{V}_l(C_i)$, the graph $\mathcal{V}_l(C_{i+1})$ is obtained by applying Lemma 15.

Let us point out a critical case, which clearly differentiates from [5]: both endpoints of $\alpha'_{i+1}$ lie on the same arc $\omega$ of $C_i$, see Figure 12(c) where $\alpha$ stands for $\alpha'_{i+1}$. That is, the insertion of $\alpha_{i+1}$ splits the arc $\omega$ in two arcs, $\omega_1$ and $\omega_2$ (note $s_{\alpha_{i+1}} <_p s_\omega$) Because of this split, $C_i$, and thus $\mathcal{V}_l(C_i)$, is order-dependent: if $\alpha_{i+1}$ were considered before $\omega$, in some alternative ordering, then $\omega_1$ or $\omega_2$ would not exist in the resulting cycle, and similarly for their faces in $\mathcal{V}_l(C_{i+1})$. The time to split $R(\omega, C_i)$ is proportional to the minimum complexity of $R(\omega_1, C_{i+1})$ and $R(\omega_2, C_{i+1})$, which is added to the time complexity of step $i$. Another side effect of the split relation is that $\alpha_{i+1}$ might fall outside $C_i$, unless $C$ is a Voronoi-cycle, in which case $C_{i+1} = C_i$. Then $C_n \neq C$, in particular, $C_n$ is enclosed by $C$.

Because the computed cycles are order-dependent, standard backwards analysis cannot be directly applied to step $i$. In [10] an alternative technique was proposed, which can be applied to the above construction. The main difference from [10] is the case $C_{i+1} = C_i$, however, such a case has no effect to time complexity, thus, the analysis of [10] can be applied.

▶ **Proposition 19.** *By the variant of backwards analysis in [10], the time complexity of step $i$ is expected $O(1)$.*

## 4.1 Relations among the Voronoi-like graphs $\mathcal{V}_l(C)$, $\mathcal{V}_l(C')$, and $\mathcal{V}_l(\hat{C})$

In the following proposition, the first claim follows from Theorem 6 and the second follows from the proof of Lemma 17.

▶ **Proposition 20.** *Let $C'$ be a Voronoi-like cycle between $C$ and $\hat{C}$ such that $S_{\hat{C}} \subseteq S_{C'} \subseteq S_C$.*

1. *$R(\alpha, C') \supseteq R(\alpha, \hat{C})$, for any arc $\alpha \in C' \cap \hat{C}$.*
2. *$R(\alpha, C') \subseteq R(\alpha, C)$, for any arc $\alpha \in C \cap C'$.*

Proposition 20 indicates that the faces of $\mathcal{V}_l(C')$ *shrink* as we move from the outer cycle $C$ to an inner one, until we reach the Voronoi faces of $\mathcal{V}_l(\hat{C})$, which are contained in all others. It also indicates that $\mathcal{V}_l(C)$, $\mathcal{V}_l(C')$ and $\mathcal{V}_l(\hat{C})$ share common subgraphs, and that the adjacencies of the Voronoi diagram $\mathcal{V}_l(\hat{C})$ are preserved. More formally,

▶ **Definition 21.** *Let $\mathcal{V}_l(C', C \cap C')$ be the following subgraph of $\mathcal{V}_l(C')$: vertex $v \in \mathcal{V}_l(C')$ is included in $\mathcal{V}_l(C', C \cap C')$, if all three faces incident to $v$ belong to arcs in $C \cap C'$; edge $e \in \mathcal{V}_l(C')$ is included to $\mathcal{V}_l(C', C \cap C')$ if both faces incident to $e$ belong to arcs in $C \cap C'$.*

▶ **Proposition 22.** *For any Voronoi-like cycle $C'$, enclosed by $C$, where $S_{C'} \subseteq S_C$, it holds: $\mathcal{V}_l(C', C \cap C') \subseteq \mathcal{V}_l(C)$.*

Depending on the problem at hand, computing $\mathcal{V}_l(C')$ (instead of the more expensive task of computing $\mathcal{V}_l(C)$) may be sufficient. For an example see [14, Section 5].

Computing $\mathcal{V}_l(C)$ in linear expected time (instead of $\mathcal{V}_l(C')$) is possible if the faces of $\mathcal{V}_l(C)$ are Voronoi regions. This can be done by deleting the superflous arcs in $C' \setminus C$, which are called *auxiliary arcs* (created by arc-splits). A concrete example is given in Section 6. During any step of the construction, if $R(\alpha', C_i)$ is a Voronoi region, but $\alpha' \cap C = \emptyset$, we can call the site-deletion procedure of [8] to eliminate $\alpha'$ and $R(\alpha', C_i)$ from $\mathcal{V}_l(C_i)$. In particular,

▶ **Proposition 23.** *Given $\mathcal{V}_l(C_i)$, $1 \leq i \leq n$, we can delete $R(\alpha, C_i)$, if $R(\alpha, C_i) \subseteq VR(s_\alpha, S_\alpha)$, where $S_\alpha \subseteq S_C$ is the set of sites that define $\partial R(\alpha, C_i)$, in expected time linear on $|S_\alpha|$.*

There are two ways to use Proposition 23, if applicable:

1. Use it when necessary to maintain the invariant that $C_i$ encloses $C$ (by deleting [8] any auxiliary arc in $C_{i-1}$ that blocks the insertion of $\alpha_i$, thus, eliminating the case $C_i = C_{i-1}$).
2. Eliminate any auxiliary arc at the time of its creation. If the insertion of $\alpha_i$ splits an arc $\omega \in C_{i-1}$ into $\omega_1$ and $\omega_2$, but $\omega_2 \notin C$, then eliminate $R(\omega_2, C_i)$ by calling [8].

The advantage of the latter is that Voronoi-like cycles become order-independent, therefore, backwards analysis can be applied to establish the algorithm's time complexity. We give the backwards analysis argument on the concrete case of Section 6; the same type of argument, only more technical, can be derived for this abstract formulation as well.

## 5 Extending to Voronoi-like cycles of $k$ sites

Theorem 13 can be extended to a *Voronoi-like $k$-cycle*, for brevity, a *$k$-cycle*, which involves a set $P$ of $k$ sites whose labels appear in the interior of the cycle. A $k$-cycle $C_k$ lies in the arrangement $\mathcal{A}(\mathcal{J}_P \cup \Gamma)$ and its vertices are degree-2 locally Voronoi, where $\mathcal{J}_P$ is the set of bisectors related to the sites in $P$. It implies a Voronoi-like graph $\mathcal{V}_l(C_k)$ involving the set of sites $S_C \subseteq S \setminus P$, which (together with the sites in $P$) define the bisector arcs of $C_k$. The definition of $\mathcal{V}_l(C_k)$ is analogous to Def. 12.

There are two types of $k$-cycles on $\mathcal{A}(\mathcal{J}_P \cup \Gamma)$ of interest: 1. *$k$-site Voronoi-like cycles* whose vertices are all of the nearest type, e.g., the boundary of the union of $k$ neighboring Voronoi regions; and 2. *order-$k$ Voronoi-like cycles* whose vertices are both of the nearest and the farthest type, e.g., the boundary of an order-$k$ Voronoi face. In either case we partition a $k$-cycle $C_k$ into *maximal compound arcs*, each induced by one site in $S_C$. Vertices in the interior of a compound arc are switches between sites in $P$, and the endpoints of compound arcs are switches between sites in $S_c$. For an order-$k$ cycle, the former vertices are of the farthest type, whereas the latter (endpoints of compound arcs) are of the nearest type.

▶ **Lemma 24.** *Assuming that it exists, $\mathcal{V}_l(C_k)$ is a forest, and if $C_k$ is bounded, then $\mathcal{V}_l(C_k)$ is a tree. Each face of $\mathcal{V}_l(C_k)$ is incident to exactly one compound arc $\alpha$ of $C_k$, which is denoted as $R(\alpha, C_k)$.*

The remaining claims of Theorem 13 can be derived similarly to Section 4, see [14].

## 6 Updating a constraint Delaunay triangulation

We give an example of a Voronoi-like cycle $C$, which does not correspond to a Voronoi region, but we need to compute the adjacencies of the Voronoi-like graph $\mathcal{V}_l(C)$. The problem appears in the incremental construction of a *constraint Delaunay triangulation* (CDT), a well-known variant of the Delaunay triangulation, in which a given set of segments is constrained to appear in the triangulation of a point set $Q$, which includes the endpoints of the segments, see [16] and references therein.

Every edge of the CDT is either an input segment or is *locally Delaunay* (see Section 1). The incremental construction to compute a CDT, first constructs an ordinary Delaunay triangulation of the points in $Q$, and then inserts segment constraints, one by one, updating the triangulation after each insertion. Shewchuk and Brown [16] gave an expected linear-time algorithm to perform each update. Although the algorithm is summarized in a pseudocode, which could then be directly implemented, the algorithmic description is quite technical having to make sense of self-intersecting polygons, their triangulations, and other exceptions. We show that the problem corresponds exactly to computing (in dual sense) the Voronoi-like graph of a Voronoi-like cycle. Thus, a very simple randomized incremental construction, with occasional calls to Chew's algorithm [5] to delete a Voronoi region of points, can be derived. Quoting from [16]: incremental segment insertion is likely to remain the most used CDT construction algorithm, so it is important to provide an understanding of its performance and how to make it run fast. We do exactly the latter in this section.

When a new constraint segment $s$ is inserted in a CDT, the triangles, which get destroyed by that segment, are identified and deleted [16]. This creates two *cavities* that need to be re-triangulated using *constrained Delaunay triangles*, see Figure 14(a),(b), borrowed from [16], where one cavity is shown shaded (in light blue) and the other unshaded. The boundary of each cavity need not be a simple polygon. However, each cavity implies a Voronoi-like cycle, whose Voronoi-like graph re-triangulates the cavity, see Figure 14(c),(d).

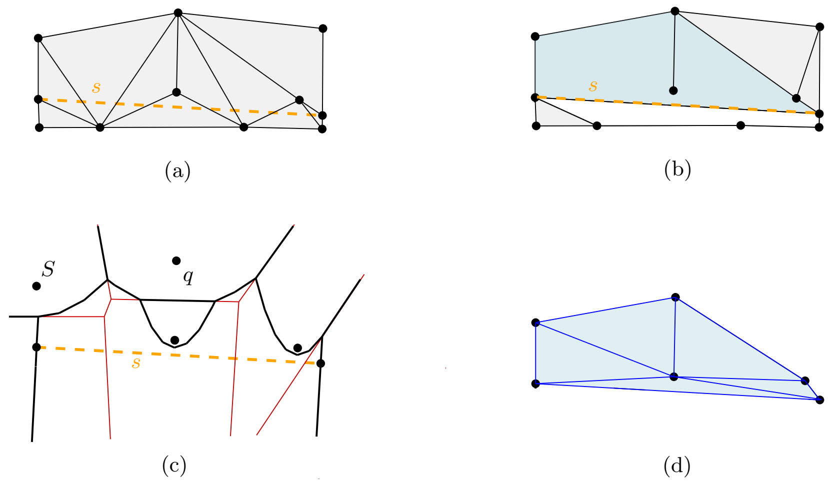



**Figure 14** Point set from [16, Fig. 3]. (a) The given CDT and the segment $s$ superimposed; (b) the cavity $P$ in blue; (c) $\mathcal{V}_l(C)$ in red, $C = \partial\mathrm{VR}(s, S \cup \{s\})$ in black; (d) the re-triangulated $P$.

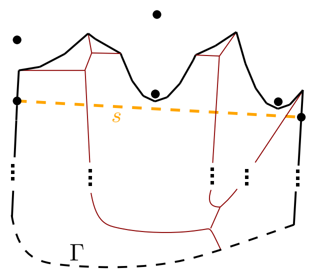



**Figure 15** The Voronoi-like cycle $C$ and $\mathcal{V}_l(C)$ (in red) for the example of Fig. 14.

Let $P = (p_1, p_2, \ldots, p_n)$ denote one of the cavities, where $p_1 \ldots p_n$ is the sequence of cavity vertices in counterclockwise order, and $p_1, p_n$ are the endpoints of $s$. Let $S$ denote the corresponding set of points ($|S| \leq n$) and let $\mathcal{J}_s$ denote the underlying bisector system involving the segment $s$ and points in $S$. Let $C$ be the $s$-cycle in $\mathcal{A}(\mathcal{J}_S \cup \Gamma)$ that has one $s$-bisector arc for each vertex in $P$, in the same order as $P$, see Figure 15. Note that one point in $S$ may contibute more than one arc in $C$.

▶ **Lemma 25.** *The $s$-cycle $C$ exists and can be derived from $P$ in linear time.*

**Proof.** Let $p_i \in P, 1 < i < n$. The diagonal of the original CDT, which bounded the triangle incident to $p_i p_{i-1}$ (resp. $p_i p_{i+1}$) was a locally Delaunay edge intersected by $s$. Thus, there is a circle through $p_i$ that is tangent to $s$ that contains neither $p_{i-1}$ nor $p_{i+1}$, see Figure 16.

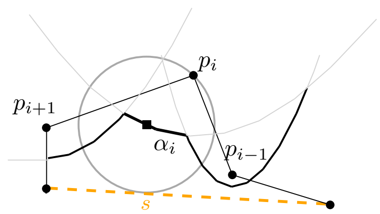



**Figure 16** Proof of Lemma 25.

Hense, an arc of $J(p_i, s)$ must exist, which contains the center of this circle, and extends from an intersection point of $J(p_i, s) \cap J(p_{i-1}, s)$ to an intersection point of $J(p_i, s) \cap J(p_{i+1}, s)$. The portion of $J(p_i, s)$ between these two intersections corresponds to the arc of $p_i$ on $C$, denoted $\alpha_i$. Note that the $s$-bisectors are parabolas that share the same directrix (the line through $s$), thus, they may intersect twice. It is also possible that $p_{i-1} = p_{i+1}$. In each case, we can determine which intersection is relavant to arc $\alpha_i$, given the order of $P$. Such questions can be reduced to *in-circle tests* involving the segment $s$ and three points. ◀

Let CDT($P$) denote the constraint Delaunay triangulation of $P$. Its edges are either locally Delaunay or they are cavity edges on the boundary of $P$.

▶ **Lemma 26.** *The CDT($P$) is dual to $\mathcal{V}_l(C)$, where $C$ is the $s$-cycle derived from $P$.*

**Proof.** The claim can be derived from the definitions, Lemma 25, which shows the existence of $C$, and the properties of Theorem 13. The dual of $\mathcal{V}_l(C)$ has one node for each $s$-bisector arc of $C$, thus, one node per vertex in $P$. An edge of $\mathcal{V}_l(C)$ incident to two locally Voronoi vertices $v, u$ involves four different sites in $S$; thus, its dual edge is locally Delaunay. The dual of an edge incident to a leaf of $C$, is an edge of the boundary of $P$. ◀

Next, we compute $\mathcal{V}_l(C)$ in expected linear time. Because $C$ is not the complete boundary of a Voronoi-region, if we apply the construction of Theorem 13, the computed cycle $C_n$ may be enclosed by $C$. This is because of occasional split operations, which may create *auxiliary arcs* that have no correspondence to vertices of $P$. However, we can use Proposition 23 to delete such auxiliary arcs and their faces. The sites in $S$ are points, thus, any Voronoi-like cycle in their bisector arrangement coincides with a Voronoi region. By calling Chew's algorithm [5] we can delete any face of any auxiliary arc in expected time linear in the complexity of the face. The side effect of always deleting auxiliary arcs is that the computed $s$-cycles are order-independent, making it possible to use backwards analysis to analyse the time complexity of step $i$, which remains $O(1)$, despite the additional calls to Chew's procedure. We give a proof and additional algorithmic details in [14].

It is easy to dualize the technique to directly compute constraint Delaunay triangles. In fact, the cycle $C$ can remain conceptual. The dual nodes are graph theoretic, each one corresponding to an $s$-bisector arc, which in turn corresponds to a cavity vertex. This explains the polygon self-crossings of [16] if we draw these graph-theoretic nodes on the cavity vertices during the intermediate steps of the construction.

## 7 Concluding remarks

We have also considered the variant of computing, in linear expected time, a Voronoi-like tree (or forest) within a simply connected domain $D$, of constant boundary complexity, given the ordering of some Voronoi faces along the boundary of $D$. In an extended version we will provide conditions under which the same essentially technique can be applied.

In future research, we are also interested in considering deterministic linear-time algorithms to compute abstract Voronoi-like trees and forests as inspired by [1].

## References

1 Alok Aggarwal, Leonidas J. Guibas, James B. Saxe, and Peter W. Shor. A linear-time algorithm for computing the Voronoi diagram of a convex polygon. *Discrete and Computational Geometry*, 4:591–604, 1989. doi:10.1007/BF02187749.

2 Franz Aurenhammer, Rolf Klein, and Der-Tsai Lee. *Voronoi Diagrams and Delaunay Triangulations*. World Scientific, 2013. URL: `http://www.worldscientific.com/worldscibooks/10.1142/8685`.

3 Cecilia Bohler, Rolf Klein, and Chih-Hung Liu. Abstract Voronoi diagrams from closed bisecting curves. *International Journal of Computational Geometry and Applications*, 27(3):221–240, 2017.

4 Cecilia Bohler, Rolf Klein, and Chih-Hung Liu. An efficient randomized algorithm for higher-order abstract Voronoi diagrams. *Algorithmica*, 81(6):2317–2345, 2019.

5 L. Paul Chew. Building Voronoi diagrams for convex polygons in linear expected time. Technical report, Dartmouth College, Hanover, USA, 1990.

6 B. Delaunay. Sur la sphère vide. A la memoire de Georges Voronoiï. *Bulletin de l'Académie des Sciences de l'URSS. Classe des sciences mathématiques et naturelles*, 6:793–800, 1934.

7 H. Edelsbrunner and R. Seidel. Voronoi diagrams and Arrangements. *Discrete and Comptational Geometry*, 1:25–44, 1986.

8 Kolja Junginger and Evanthia Papadopoulou. Deletion in Abstract Voronoi Diagrams in Expected Linear Time. In *34th International Symposium on Computational Geometry (SoCG)*, volume 99 of *LIPIcs*, pages 50:1–50:14, Dagstuhl, Germany, 2018.

9 Kolja Junginger and Evanthia Papadopoulou. On tree-like abstract Voronoi diagrams in expected linear time. In *CGWeek Young Researchers Forum (CG:YRF)*, 2019.

10 Kolja Junginger and Evanthia Papadopoulou. Deletion in abstract Voronoi diagrams in expected linear time and related problems. arXiv:1803.05372v2 [cs.CG], 2020. To appear in *Discrete and Computational Geometry.*

11 Rolf Klein. *Concrete and Abstract Voronoi Diagrams*, volume 400 of *Lecture Notes in Computer Science*. Springer-Verlag, 1989.

12 Rolf Klein, Elmar Langetepe, and Z. Nilforoushan. Abstract Voronoi diagrams revisited. *Computational Geometry: Theory and Applications*, 42(9):885–902, 2009.

13 Rolf Klein, Kurt Mehlhorn, and Stefan Meiser. Randomized incremental construction of abstract Voronoi diagrams. *Computational Geometry: Theory and Applications*, 3:157–184, 1993.

14 Evanthia Papadopoulou. Abstract Voronoi-like Graphs: Extending Delaunay's Theorem and Applications. arXiv:2303.06669 [cs.CG], March 2023. `doi:10.48550/arXiv.2303.06669`.

15 Micha Sharir and Pankaj K. Agarwal. *Davenport-Schinzel sequences and their geometric applications*. Cambridge university press, 1995.

16 Jonathan Richard Shewchuk and Brielin C. Brown. Fast segment insertion and incremental construction of constrained Delaunay triangulations. *Computational Geometry: Theory and Applications*, 48(8):554–574, 2015.