A Sparse Delaunay Filtration

Donald R. Sheehy ✉ ⌂ ⓘ
North Carolina State University, Raleigh, NC, USA

──── **Abstract** ────

We show how a filtration of Delaunay complexes can be used to approximate the persistence diagram of the distance to a point set in $\mathbb{R}^d$. Whereas the full Delaunay complex can be used to compute this persistence diagram exactly, it may have size $O(n^{\lceil d/2 \rceil})$. In contrast, our construction uses only $O(n)$ simplices. The central idea is to connect Delaunay complexes on progressively denser subsamples by considering the flips in an incremental construction as simplices in $d + 1$ dimensions. This approach leads to a very simple and straightforward proof of correctness in geometric terms, because the final filtration is dual to a $(d + 1)$-dimensional Voronoi construction similar to the standard Delaunay filtration. We also, show how this complex can be efficiently constructed.

1 Introduction

The persistent homology of the distance to a set of points P in $\mathbb{R}^d$ describes the evolution of the topology of $\bigcup_{p \in P} \text{ball}(p, \alpha)$ as α grows from 0 to ∞. It is a multi-scale description of the "shape" of the point set. The theory of persistent homology has its origins in the work by Edelsbrunner et al. [17, 15] on α-hulls and α-shapes and their relation to the Delaunay triangulation. The paper that introduced persistent homology [18] was based on ordering the simplices of the Delaunay triangulation. Many papers have followed that use alternatives to the Delaunay triangulation in different spaces, but in Euclidean space, the Delaunay triangulation has a certain perfection in its ability to represent the topology of the distance function. For approximations, the potential $O(n^{\lceil d/2 \rceil})$ size of the Delaunay triangulation cannot compete with the linear size of so-called sparse filtrations [35, 2, 6, 13]. In this paper, we combine the ideas from sparse filtrations with the Delaunay triangulation, achieving both the elegance of the Delaunay triangulation and the worst-case linear-size guarantees. Along the way, we will give a new topological perspective to the classic approach to computing Delaunay triangulations by flips.

A *flip* in a 2-dimensional triangulation is the replacement of two adjacent triangles whose four vertices are in convex position with the other two possible triangles on the same vertices. A classic way of visualizing flips is to view the two configurations as projections of the upper and lower hull of a tetrahedron in three dimensions (see Fig. 1). This view also permits one to interpret other operations as flips, such as the insertion of a new vertex splitting one triangle into three. We call the former class of flips $(2, 2)$-flips and the latter $(1, 3)$-flips, indicating the number of triangles before and after the flip. More generally, there are $(k, d + 2 - k)$-flips for sets of $d + 2$ points in $\mathbb{R}^d$. These are likewise interpreted as projections of $(d + 1)$-simplices. In this paper, we will use the $(d + 1)$-simplices of the flips to give a topological connection between the Delaunay triangulation of a set of points and the Delaunay triangulation of a subset – both triangulations are subcomplexes of a $(d + 1)$-dimensional complex containing the flip simplices. Throughout, we will distinguish between the terms Delaunay triangulation
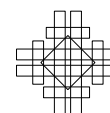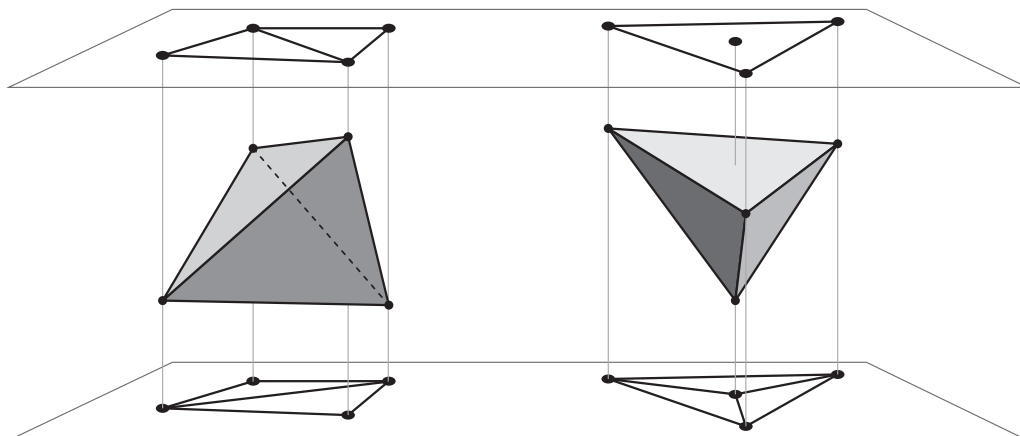
■ **Figure 1** Flips in the plane correspond to the upper and lower facets of a tetrahedron in $\mathbb{R}^3$.

for the embedded geometric complex on a set of points and a Delaunay complex, which is the corresponding abstract simplicial complex. As will become clear, the addition of the flip simplices will result in a simplicial complex that will not be embedded in $\mathbb{R}^d$.

Many Delaunay triangulation algorithms use flips as an algorithmic primitive. This idea goes back to the work of Lawson [28, 27] and reached its more modern form in the simultaneous papers of Bowyer [3] and Watson [37]. As the d-simplices of the Delaunay triangulation are those for which the circumsphere is empty of other input points, a transformations of the problem into $\mathbb{R}^{d+1}$ can make this criterion linear. This was first done by Brown [4] using the stereographic map and later by Edelsbrunner and Seidel [19] using the parabolic lifting. Adjusting the parabolic lifting can be interpreted as assigning weights to points and leads to *weighted Delaunay triangulations* (also known as *regular* triangulations). Edelsbrunner and Shah showed that incremental, flip-based algorithms also work in the weighted case [20] despite obstructions discovered by Joe [26] to applying Lawson's algorithm in $\mathbb{R}^3$.

The theory of persistent homology applies to settings much more general than the sublevel sets of distance functions in Euclidean space. One common setting is to consider points in finite metric spaces. The complex used for this is called the (Vietoris-)Rips complex. At scale α, it contains a simplex for every clique in the α-neighborhood graph of the points. In order to deal with the size blowup of this complex, several sparsification methods have been proposed [35, 2, 6, 13]. All of these approaches attempt to use only subsets of the input points as the scale increases. In this paper, we will show how to adapt this approach to the Delaunay complex.

In other related work, several authors have given sparse approximations for distance functions in Euclidean space. For example, Hudson et al. [25] use ideas from mesh generation to give an approximation to the distance function. Choudhary et al. [11] used a method of overlapping grids to give an approximation that has an asymptotically smaller dependence on the dimension than previous methods.

2 Background

2.1 Distances, Points, and Weights

Let $\|a - b\|$ denote the Euclidean distance between a and b in $\mathbb{R}^d$. Let $\mathrm{ball}(c, r)$ denote the closed ball centered at $c \in \mathbb{R}^d$ with radius r. For a set $P \subset \mathbb{R}^d$, let

$$\mathbf{d}(x, P) = \min_{p \in P} \|x - p\|.$$

Equivalently, $\mathbf{d}(x, P)$ is the minimum r such that $P \cap \mathrm{ball}(x, r)$ is nonempty.

The distance function induced by a point set P maps each point $x \in \mathbb{R}^d$ to $\mathbf{d}(x, P)$. The sublevel sets of this distance function is often used in topological data analysis as a way to extend a finite set and fill in the space between the points. These sublevel sets are sometimes called *offsets* and are formally defined for each scale α as

$$P^\alpha := \{x \in \mathbb{R}^d \mid \mathbf{d}(x, P) \le \alpha\} = \bigcup_{p \in P} \mathrm{ball}(p, \alpha).$$

The Hausdorff distance between two point sets P and Q is defined as

$$\mathbf{d}_H(P, Q) := \max\{\max_{p \in P} \mathbf{d}(p, Q), \max_{q \in Q} \mathbf{d}(q, P)\}.$$

Equivalently, $\mathbf{d}_H(P, Q)$ is the minimum r such that $P \subseteq Q^r$ and $Q \subseteq P^r$.

One way to modify a distance function and give more or less importance to certain points is to assign a weight to each point. A weighted point $\hat{p}$ is a point $p \in \mathbb{R}^d$ and a weight $w_p \in \mathbb{R}$. The *weighted distance to $\hat{p}$* is defined as

$$\pi_p(x) := \sqrt{\|x - p\|^2 + w_p^2}.$$

The weighted distance is also called the power distance, especially in the case where one subtracts the weight rather than adds it. Throughout the paper, we add weights rather than the usual power distance as it substantially simplifies both the conceptual use of weights to decrease the importance (i.e., radius) of some points and also the arithmetic. All of the constructions we present can be translated into power distances by a global transformation of all the weights and a reinterpretation of the scale. A similar approach to weighting point may be found in [5].

We use the same notation for the weighted distance of a (non-weighted) point x to a weighted point set $\hat{P}$ as we did in the unweighted case.

$$\mathbf{d}(x, \hat{P}) := \min_{\hat{p} \in \hat{P}} \pi_{\hat{p}}(x).$$

Thus, unweighted points way be viewed as points with weight zero. The offsets of $\hat{P}$ are

$$\hat{P}^\alpha = \{x \in \mathbb{R}^d \mid \mathbf{d}(x, \hat{P}) \le \alpha\}.$$

2.2 Persistent Homology

A family of sets $\{X^\alpha \mid \alpha \in \mathbb{R}\}$ is called a *filtration* if for all $\alpha \le \beta$, we have $X^\alpha \subseteq X^\beta$. We will reserve superscripts on sets as a notation for filtration parameters and will denote a filtration (X^α) with parentheses to stress the importance of the ordering. For filtrations that are defined only over an interval $[s, t] \subset \mathbb{R}$, we will assume that $X^\alpha = X^s$ for $\alpha < s$ and $X^\alpha = X^t$ for $\alpha > t$.

The *persistent homology* of (X^α) is a representation of the changes in the topology of X^α as α varies over $\mathbb{R}$. The result is a *persistence diagram*, denoted $\text{Dgm}(X^\alpha)$, that is a multiset of pairs (b, d) in the extended plane $(\mathbb{R} \cup \infty)^2$. Each pair (b, d) represents a nontrivial homology class that exists only in X^α for α in the half open interval $[b, d)$. Thus, b is the *birth time* of a topological feature and d is its *death time*.

Persistent homology is usually computed on combinatorial objects called simplicial complexes. A *simplicial complex* is a pair of sets (V, S) where V is the vertex set and $S \subseteq \text{Pow}(V)$ is the simplex set. It is required that S is closed under subsets, i.e., if $\sigma \subseteq \tau \in S$, then $\sigma \in S$. A family (K^α) of simplicial complexes is called a *filtered simplicial complex* if for all $\alpha \leq \beta$, we have K^α is a subcomplex of K^β.

The main problem addressed in this paper is the efficient approximation of $\text{Dgm}(P^\alpha)$ by constructing a linear-size filtered simplicial complex based on the Delaunay triangulation.

2.3   Greedy Permutations

For ranges of indices, let $[a : b]$ denote $\{a, \ldots b-1\}$ and $[b]$ denote $[0 : b]$. If the input set P is ordered as $(p_0, \ldots, p_{n-1})$, let $P_i\{p_j \mid 0 \leq j < i\}$ denote the ith prefix of the ordering. The *spread* Δ of P is the ratio of the largest and smallest pairwise distances among points of P.

A *greedy ordering* or *greedy permutation* of P is defined as follows. The first point, p_0, may be chosen arbitrarily. The ith point, p_i is chosen to be a point that maximizes $\mathbf{d}(p_i, P_i)$. That is, each point after the first is the farthest from its predecessors. Equivalently,

$$\mathbf{d}(p_i, P_i) = \mathbf{d}_H(P_i, P).$$

Greedy permutations have been reinvented several times, especially in the context of k-center clustering (see Gonzalez [21] or Dyer and Frieze [14]). Clarkson [12] adapted his sb data structure for nearest neighbor search to compute greedy permutations. Har-peled and Mendel [23] showed that Clarkson's approach yields an $O(n \log \Delta)$-time algorithm in low-dimensional metric spaces. They also gave an $O(n \log n)$-time algorithm in such cases.

The *insertion radius* of a point p_i is defined as

$$r_i := \mathbf{d}(p_i, P_i)$$

By convention, $r_0 = \infty$. For greedy orderings, if $i < j$, then $r_i \geq r_j$.

Every prefix P_i of a greedy permutation satisfies both packing and covering conditions in the following sense. The set P_i is a r_{i-1}-*packing*: for every pair of points in $a, b \in P_i$, we have $\|a - b\| \geq r_{i-1}$. The set P_i is an r_i-*covering* of P: for every point $a \in P$, there is a point $b \in P_i$ such that $\|a - b\| \leq r_i$.

2.4   Weights and Time

We will be considering weighted point sets in which the weights vary in time. For each point p_i in P, we will assign a nonnegative weight function $w_i : \mathbb{R} \to \mathbb{R}$. For a given α, the set

$$\hat{P}(\alpha) := \{(p_i, w_i(\alpha)) \mid p_i \in P\}$$

is a weighted point set in which the weight of p_i is $w_i(\alpha)$. Let $\pi_{i,\alpha}(x)$ denote the weighted distance from the weighted point $(p_i, w_i(\alpha))$ to x.

The weight functions will be defined in terms of the *freezing time* λ_i of each point p_i. The exact value chosen for λ_i will depend on our desired approximation guarantees and the specifics of the algorithm. Once the freezing times are fixed, the weights are as follows.

$$w_i(\alpha) = \begin{cases} 0 & \text{if } \alpha < \lambda_i \\ \sqrt{\alpha^2 - \lambda_i^2} & \text{otherwise} \end{cases}$$

The ball $b_i(\alpha)$ centered at p_i with weight $w_i(\alpha)$ is

$$b_i(\alpha) = \{x \in \mathbb{R}^d \mid \pi_{i,\alpha}(x) \leq \alpha\} = \text{ball}(p_i, \min\{\alpha, \lambda_i\}).$$

This is why λ_i is called the freezing time; at scales $\alpha > \lambda_i$, the Euclidean radius of a ball of weighted radius α will not grow.

The following lemma shows how weighting the points according to the freezing time guarantees that at all scales, there is always a point nearby that is sufficiently close and sufficiently far from its freezing time. The proof can be found in the full version.

▶ **Lemma 1.** *Let $P \subset \mathbb{R}^d$ be ordered according to a greedy permutation with insertion radii $r_0 \ldots r_{i-1}$. Let $\varepsilon > 0$ be any constant. For all $j \in [1:n]$, let the freezing times λ_j be chosen so that $\lambda_j \geq \frac{1+\varepsilon}{\varepsilon} r_j$. Then, for all $k \in [1:n]$ and all $\alpha \geq 0$, there exists i such that*
- $\lambda_i \geq (1 + \varepsilon)\alpha$, *and*
- $\|p_i - p_k\| < \varepsilon\alpha$.

In the special case of $\alpha = \lambda_k$ in the preceding lemma, $\text{ball}(p_k, \lambda_k)$ is completely covered by $\text{ball}(p_i, (1 + \varepsilon)\lambda_k)$, where the point p_i is not yet frozen at time $(1 + \varepsilon)\lambda_k$. For larger values of α, there will always be a point p_i to cover this ball. This lemma has two important consequences. First, it implies that we will be able to remove or ignore the point p_k after time $(1 + \varepsilon)\lambda_k$. Second, it allows us to relate the offsets of P with the weighted offsets as shown in the following lemma whose proof may be found in the full paper.

▶ **Lemma 2** (Weighted Offset Interleaving). *Let $P \subset \mathbb{R}^d$ be ordered according to a greedy permutation with insertion radii $r_0 \ldots r_{i-1}$. Let $\varepsilon > 0$ be any constant. For all $j \in [1:n]$, let the freezing times λ_j be chosen so that $\lambda_j \geq \frac{1+\varepsilon}{\varepsilon} r_j$. Then, for all $\alpha \geq 0, \hat{P}^\alpha \subseteq P^\alpha \subseteq \hat{P}^{(1+\varepsilon)\alpha}$.*

2.5 Delaunay and Voronoi
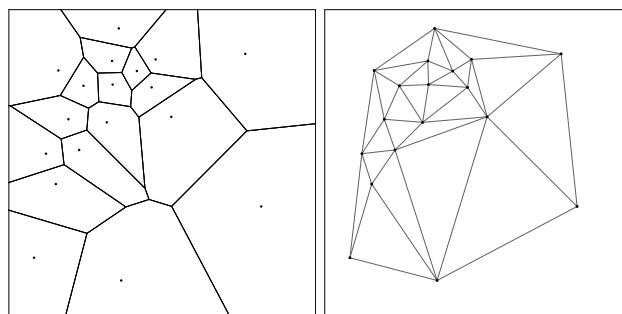
Figure 2 The Voronoi diagram and its dual Delaunay triangulation.

Let P be a set of points in $\mathbb{R}^d$. The *Voronoi cell* of a point $q \in P$ is defined as

$$\text{Vor}_P(q) := \{x \in \mathbb{R}^d \mid \|x - q\| = \mathbf{d}(x, P)\}.$$

For a subset of points $S \subseteq P$, we can define its Voronoi cell as

$$\mathrm{Vor}_P(S) := \bigcap_{q \in S} \mathrm{Vor}_P(q).$$

The same definitions apply to weighted points. For a given α, the Voronoi cell of q is

$$\mathrm{Vor}_{\hat{P}(\alpha)}(q) := \{x \in \mathbb{R}^d \mid \pi_q(x) = \pi(x, \hat{P}(\alpha))\}.$$

It is well-known that the Voronoi cells are polyhedra. The *Voronoi diagram* of P is defined as the polyhedral complex composed of nonempty Voronoi cells $\mathrm{Vor}_P(S)$ for all $S \subseteq P$.

The *Delaunay complex* (also known with some nuances as the Delaunay triangulation, tesselation, or mosaic) is the simplicial complex formed by the subsets $S \subseteq P$ for which $\mathrm{Vor}_P(S)$ is nonempty. The subsets are the *simplices* and the *dimension* of S is $|S| - 1$. We are defining the Delaunay complex here as an abstract simplicial complex. In the special case where all simplices have dimension at most d, the Delaunay complex will embed neatly into $\mathbb{R}^d$ with the vertices embedded at the points of P and each simplex embedded as the convex closure of its vertices. For the purposes of this paper we will not need the embedding and thus will have no need for the usual general position conditions as would usually be required for the geometric realization of the complex in $\mathbb{R}^d$. In fact, we will explicitly construct "degenerate" Delaunay complexes because the adjustment of weights over time will necessarily pass through instants where higher dimensional simplices are present in the Delaunay complex. These are the moments when a flip occurs. We will only require that at most one flip occurs at a time.

2.6   The Kinetic View of Flips

Kinetic data structures [22] generalize the classic sweepline approach of Bentley and Ottmann. The goal is to maintain some geometric structure as points move along trajectories. The principal technique is to rewrite the geometric predicates defining the structure as functions (usually polynomials) of time and then solving (finding roots) for the time when the predicate will no longer hold. At that time, some combinatorial change is required. These changes are stored in a priority queue, ordered by time.

Incremental Delaunay triangulation can be phrased as a kinetic data structures problem if one understand the motion in $d + 1$ dimensions as a continuous change in the weight of the point being inserted. This perspective is often abandoned because the precise ordering of the flips is rarely important and is not necessary for correct computation (see Edelsbrunner [20]). However, in our case, we want the precise order and time of the flips that occur, because these inform the final filtration. Also, we will be adding multiple points at once, so the order is important.

A similar approach was used by Miller and Sheehy [30] in an output-sensitive algorithm for computing Delaunay triangulations. In that paper, it was observed that the predicate polynomials are linear for the case where the points are partitioned into two sets, one with weight zero and one with squared weight varying linearly in time.
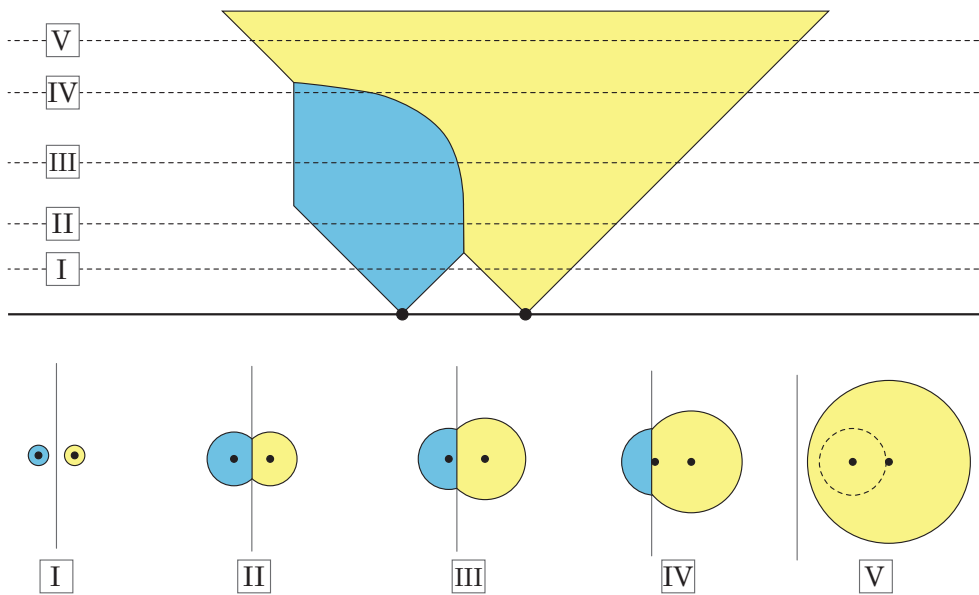
2.7   Clipping the Voronoi Diagram

The *clipped Voronoi cell* is the intersection of a Voronoi cell and a ball. Let $\hat{P}$ be a weighted point set with weights varying in time as above. Then, for each $p_i \in \hat{P}$ and each $\alpha \geq 0$ there is a ball

$$b_i(\alpha) := \mathrm{ball}(p_i, \min\{\lambda_i, \alpha\}).$$
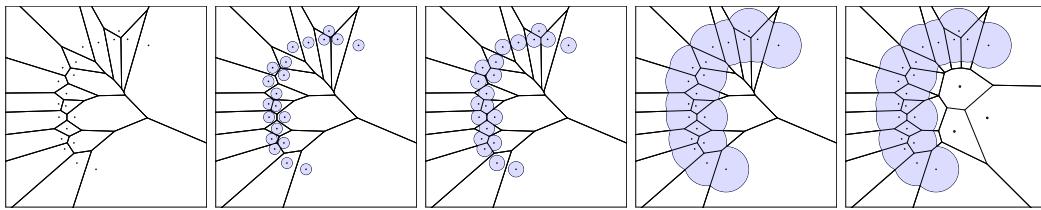
Figure 3 The clipped Voronoi cells exactly cover the offsets. In the last frame, we illustrate how the addition of extra points may not change the offsets or the Delaunay filtration. These extra points are used in Section 4 to keep the complexity linear.

Figure 4 For two points, the lifted, clipped Voronoi cells are shown from the side. The clipped Voronoi diagrams at five different scales are illustrated. Note that at some scale, the left ball ceases growing. Then, it is overtaken by the cell of the right point.

Specifically, for $p_i \in P$, we define the clipped Voronoi cell of p_i as

$$V_i^\alpha := \mathrm{Vor}_{\hat{P}(\alpha)}(p_i) \cap b_i(\alpha).$$

The *Delaunay complex at scale* α is the subcomplex of the Delaunay complex defined by using the clipped Voronoi cells instead of the full Voronoi cells. That is,

$$D^\alpha := \{\sigma \subseteq P \mid \bigcap_{p_i \in \sigma} V_i^\alpha \neq \emptyset\}.$$

By defining the weights as above, we guarantee that for all scales $\alpha \geq (1 + \varepsilon)\lambda_k$, the clipped Voronoi cell V_k^α will be empty. This is a direct consequence of Lemma 1, but we give the formal statement below and the proof in the full version for completeness. (See Fig. 4.)

▶ **Lemma 3.** *Let $P \subset \mathbb{R}^d$ be ordered according to a greedy permutation with insertion radii $r_0 \ldots r_{i-1}$. Let $\varepsilon > 0$ be any constant. For all $j \in [1 : n]$, let the freezing times λ_j be chosen so that $\lambda_j \geq \frac{1+\varepsilon}{\varepsilon} r_j$. Then, for all $k \in [1 : n]$ and all $\alpha > (1 + \varepsilon)\lambda_k$, we have $V_k^\alpha = \emptyset$.*

This is how some points will cease to impact the filtration at larger scales. In the next section, we will see how to simulate the removal of vertices whose clipped Voronoi cells are empty. Before we give that construction, we will relate the clipped Voronoi diagram to the (weighted) offsets. For completeness, we include a proof of this well-known fact (see for example Edelsbrunner [15]) in the full version.

▶ **Lemma 4.** *Let $P \subset \mathbb{R}^d$ be ordered according to a greedy permutation with insertion radii $r_0 \dots r_{i-1}$. Let $\varepsilon > 0$ be any constant. For all $j \in [1:n]$, let the freezing times λ_j be chosen so that*

$$\lambda_j \geq \frac{1+\varepsilon}{\varepsilon} r_j.$$

Then, for all $\alpha \geq 0$,

$$\hat{P}^\alpha = \bigcup_{i \in [n]} V_i^\alpha.$$

2.8   Extending Voronoi and Delaunay in Time and Space

Given that we will be considering weighted point sets in which the weights of the points vary in time, it is useful to give a concrete geometric structure that captures the evolution of the Voronoi diagram and the Delaunay triangulation.

Define

$$\hat{P}_+^\alpha := \{(x, \gamma) \in \mathbb{R}^d \times [0, \alpha] \mid x \in \hat{P}^\gamma\}.$$

This filtration has the property that for any $\alpha \geq 0$, there is a natural homotopy equivalence $m : \hat{P}_+^\alpha \to \hat{P}^\alpha$ defined by the projection $m(x, \gamma) = x$. This follows from the fact that $\hat{P}^\alpha$ only grows with increasing α and thus the fibers of m are simply connected (line segments).

We can similarly embed the clipped Voronoi cells in $\mathbb{R}^{d+1}$ by defining

$$V_{+i}^\alpha := \{(x, \gamma) \in \mathbb{R}^d \times [0, \alpha] \mid x \in V_i^\gamma\}.$$

The collection of these sets is

$$V_+^\alpha := \{V_{+i}^\alpha\}_{i \in [n]}.$$

The extended Delaunay complex is defined as

$$D_+^\alpha := \{\sigma \subseteq P \mid \bigcap_{p_i \in \sigma} V_{+i}^\alpha \neq \emptyset\}.$$

So, D_+^α contains every simplex that appears in any Delaunay complex D^γ for any $\gamma \in [0, \alpha]$.

2.9   Nerves

Construction of the Delaunay triangulation from the Voronoi cells is an example of a *nerve*. More generally, given a collection U of sets, we define a simplicial complex

$$\mathrm{Nrv}(U) := \{\sigma \subset U \mid \bigcap_{S \in \sigma} S \neq \emptyset\}.$$

For the Delaunay triangulation, if there is a nonempty intersection of Voronoi cells, we identify the corresponding simplex with the set of points defining those cells. So, D^α is (isomorphic to) the nerve of V^α and D_+^α is the nerve of V_+^α.

The collection U is called a *cover*, and the set $W = \bigcup_{S \in U} S$ is the set that U covers. A cover is a *good cover* if the intersections of elements are all either empty or contractible. In the case of the Voronoi diagram as well as the clipped Voronoi diagram, the convexity of the cells guarantees that the cover is good.

The *Nerve Theorem* says that the nerve of a good cover is homotopy equivalent to the union, so the homology of the nerve matches the homology of the union. In the case of the clipped Voronoi diagram, the Nerve Theorem implies that the Delaunay complex at scale α is equivalent in homology to the weighted offsets. This observation was one of the main ideas that drove the development of persistent homology.

If, instead of a collection of sets, we have a collection of filtrations, we can define their nerve as a filtered simplicial complex. For example, if we have filtrations $\{(U_0^\alpha), \dots, (U_{k-1}^\alpha)\}$, then for each α we can define the cover $U^\alpha = \{U_0^\alpha, \dots, U_{k-1}^\alpha\}$ of $W^\alpha = \bigcup_{i \in [k]} U_i^\alpha$ and the nerve $N^\alpha = \mathrm{Nrv}(U^\alpha)$. Then, we have a new filtration (W^α) and a filtered simplicial complex (N^α). The Persistent Nerve Lemma [8] implies that if U^α is a good cover for all α, then the persistence diagrams of (N^α) and (W^α) are identical.

3 The Sparse Delaunay Filtration

In this section, we will prove that our sparse Delaunay filtration $\mathrm{Dgm}(D_+^\alpha)$ is a good approximation to $\mathrm{Dgm}(P^\alpha)$. In Section 2, we established the geometric lemmas that will allow us to relate $\mathrm{Dgm}(P^\alpha)$ with $\mathrm{Dgm}(\bigcup V_+^\alpha)$. We will use the Persistent Nerve Lemma to show that $\mathrm{Dgm}(\bigcup V_+^\alpha) = \mathrm{Dgm}(D_+^\alpha)$, but it will require showing that lifted clipped Voronoi cells form a good cover. Although the slices of these cells are convex at any fixed α, they are not themselves convex as can be seen in the example in Fig. 4. In this section, we will show despite having non-convex sets, V_+^α *is* a good filtered cover. The proof will depend on our careful choice of weights.

3.1 Adding Points in Waves

To define the weight functions, it suffices to establish the freezing times. The specific choice of the freezing times impacts the size (larger freezing times yields a larger filtration), but also the correctness (the cover must be good). For many of the preceding lemmas, it was necessary to choose freezing times so that

$$\lambda_i \geq \frac{1+\varepsilon}{\varepsilon} r_i,$$

where r_i is the insertion radius of p_i in a greedy ordering of the input set P. Recall that ε is the user chosen parameter that will define the accuracy of our approximation.

We will satisfy this requirement by grouping points according to their insertion radius, rounding up to the nearest power of $1 + \varepsilon$. All the points in a group will have the same freezing time. Formally, we set

$$\lambda_i := (1+\varepsilon)^{\left\lceil \log_{1+\varepsilon}\left(\frac{1+\varepsilon}{\varepsilon} r_i\right) \right\rceil}.$$

Rounding the weights simplifies the proofs, but it is an open question as to whether it is necessary for the correctness of the construction.

The filtration (D_+^α) defined with these weight functions on a greedy permutation is *The Sparse Delaunay Filtration*.

3.2   Monotonicity

The first step in showing that V_+^α is a good filtered cover is to show that every Delaunay simplex in D^α appears and disappears at most once. There are two ways that a simplex can be removed at time α. First, it may be that its Voronoi cell becomes empty and therefore it is removed from the Delaunay triangulation entirely. This is the standard case to analyze in flip-based Delaunay computation. Second, it may be that only the clipped Voronoi cell becomes empty. In this case, we must show that it remains empty for the rest of the filtration.

The challenge is that the lifted Voronoi cells are not convex. Aurenhammer et al. [1] showed that this monotonicity does not hold for a related set of weight functions. That paper addressed the problem of computing the Voronoi diagram of parallel halflines and showed that the slices perpendicular to the halflines are weighted Voronoi diagrams. Similar to the current paper, they construct a $d + 1$-dimensional decomposition by sweeping through d-dimensional slices. That paper gives an example attributed to Peter Widmayer's research group in which a particular triangle would be flipped out and later flipped back in. Such a non-monotone example highlights the need to be careful in choosing the weights.

▶ **Lemma 5.** *If $\sigma \in D^\alpha$ and $\sigma \in D^\beta$, then $\sigma \in D^\gamma$ for all $\gamma \in [\alpha, \beta]$.*

Proof. Let $\lambda_\sigma = \min_{p_i \in \sigma} \lambda_i$. Lemma 3 implies that no point p_i appears in D^γ for $\gamma > (1+\varepsilon)\lambda_i$. So, it must be that $\beta \le (1 + \varepsilon)\lambda_\sigma$. If $\alpha < \lambda_\sigma$, the points p_i of σ all have weight $w_i(\gamma) = 0$ for all $\gamma \in [\alpha, \lambda_\sigma]$. In that case, the nonempty set $\bigcap_{p_i \in \sigma} V_i^\gamma$ is a subset of $\bigcap_{p_i \in \sigma} V_i^{\lambda_\sigma}$ and so, $\sigma \in D^\gamma$. So, it will suffice to prove the lemma for the case where α and β are in the interval $[\lambda_\sigma, (1 + \varepsilon)\lambda_\sigma]$.

For any $\gamma \in [\alpha, \beta]$, let $t = \frac{\gamma^2 - \alpha^2}{\beta^2 - \alpha^2}$.

This choice implies that $\gamma = \sqrt{(1 - t)\alpha^2 + t\beta^2}$.

By rounding the freezing times to powers of $(1 + \varepsilon)$, there are no freezing times in the open interval (α, β). In particular, $w_i(\alpha) = 0$ iff $w_i(\beta) = 0$. So, for all $p_i \in P$, we have

$$w_i(\gamma)^2 = (1 - t)w_i(\alpha)^2 + tw_i(\beta)^2.$$

Let x be a point in the intersection $\bigcap_{p_i \in \sigma} V_i^\alpha$ and let y be a point in $\bigcap_{p_i \in \sigma} V_i^\beta$. These points witness the existence of σ in D^α and D^β respectively. We will show that $z = (1-t)x + ty$ is in $\bigcap_{p_i \in \sigma} V_i^\gamma$.

Let $p_i \in \sigma$ be chosen arbitrarily. Using the convexity of the squared distance to a point,

$$
\begin{aligned}
\pi_{i,\gamma}(z)^2 &= \|p_i - z\|^2 + w_i(\gamma)^2 \\
&= \|p_i - z\|^2 + (1 - t)w_i(\alpha)^2 + tw_i(\beta)^2 \\
&\le (1 - t)\|p_i - x\|^2 + t\|p_i - y\| + (1 - t)w_i(\alpha)^2 + tw_i(\beta)^2 \\
&= (1 - t)\pi_{i,\alpha}(x) + t\pi_{i_\beta}(y) \\
&\le (1 - t)\alpha^2 + t\beta^2 \\
&= \gamma.
\end{aligned}
$$

So, $z \in b_i(\gamma)$.

Let $p_j \in P$ be any point. Because $x \in \mathrm{Vor}_{\hat{P}(\alpha)}(\sigma)$ and $y \in \mathrm{Vor}_{\hat{P}(\beta)}(\sigma)$, we know that

$$\pi_{j,\alpha}(x)^2 \ge \pi_{i,\alpha}(x)^2, \text{ and } \pi_{j,\beta}(y)^2 \ge \pi_{i,\beta}(y)^2.$$

So, by the definition of the power distance,

$$
\begin{aligned}
\pi_{j,\gamma}(z)^2 - \pi_{i,\gamma}(z)^2 &= \|p_j - z\|^2 - \|p_i - z\|^2 + w_j(\gamma)^2 - w_i(\gamma)^2 \\
&= \|p_j\|^2 - \|p_i\|^2 - 2z^\top(p_j - p_i) + w_j(\gamma)^2 - w_i(\gamma)^2 \\
&= (1-t)(\pi_{j,\alpha}(x) - \pi_{i,\alpha}(x)) + t(\pi_{j,\beta}(y) - \pi_{i,\beta}(y)) \\
&\geq 0.
\end{aligned}
$$

So, for all $p_i \in \sigma$ and $p_j \in P$, we have $\pi_{i,\gamma}(z) \leq \pi_{j,\gamma}(z)$ and thus $z \in \mathrm{Vor}_{\hat{P}(\gamma)}(p_i)$. We have shown $z \in b_i^\gamma$ and $z \in \mathrm{Vor}_{\hat{P}(\gamma)}(p_i)$, and therefore, $z \in b_i^\gamma \cap \mathrm{Vor}_{\hat{P}(\gamma)}(p_i) = V_i^\gamma$. As this holds for all $p_i \in \sigma$, we have a point $z \in \bigcap_{p_i \in \sigma} V_i^\gamma$, and thus, $\sigma \in D^\gamma$ as desired. ◄

3.3 A Good Filtered Cover

We can now prove that the lifted Voronoi cells form a good filtered cover.

▶ **Lemma 6.** *Let $\varepsilon > 0$ be any constant. Let V_+^α be the lifted Voronoi cells whose nerve is the ε-Sparse Delaunay Filtration for $P \subset \mathbb{R}^d$. Then, (V_+^α) is a good filtered cover of $(\hat{P}^\alpha)$.*

Proof. Fix any $\alpha \geq 0$. Let $\sigma \in \mathrm{Nrv}(V_+^\alpha)$ be any simplex. We will show that the intersection of the lifted, clipped Voronoi cells $\{V_i^\alpha \mid p_i \in \sigma\}$ is contractible.

In each d-dimensional slice, the clipped Voronoi cells V_i^α are convex, so their intersection is convex. We can deformation retract the nonempty clipped Voronoi cells in each slice to the orthocenter, i.e., the point in the cell that minimizes the maximum weighted distance to the points of σ. The cells change continuously in time and so does the orthocenter of the simplex. The retractions in each slice will be continuous as a retraction in $\mathbb{R}^{d+1}$. By Lemma 5 the collection of orthocenters in the slices will form a connected path and thus are contractible. ◄

▶ **Theorem 7.** *Let $\varepsilon > 0$ be any constant. Let (D_+^α) be the ε-Sparse Delaunay Filtration for $P \subset \mathbb{R}^d$. Then, $\mathrm{Dgm}(D_+^\alpha)$ is a $(1+\varepsilon)$-approximation to $\mathrm{Dgm}(P^\alpha)$.*

Proof. By Lemma 6, the lifted clipped Voronoi cells $\{V_i^\alpha\}$ form a good cover of $\hat{P}^\alpha$ for all α and therefore, by the Persistent Nerve Lemma,

$$\mathrm{Dgm}(D^\alpha) = \mathrm{Dgm}(\hat{P}^\alpha).$$

Lemma 2 gives an interleaving of persistence modules (see [7]) so that $\mathrm{Dgm}(\hat{P}^\alpha)$ is a $(1+\varepsilon)$-approximation to $\mathrm{Dgm}(P^\alpha)$. Combining these facts, we get that $\mathrm{Dgm}(D^\alpha)$ is a $(1+\varepsilon)$-approximation to $\mathrm{Dgm}(P^\alpha)$. ◄

3.4 Size Analysis

In general, the Delaunay complex on n points (in general position) may have $O(n^{\lceil d/2 \rceil})$ simplices [32]. There are several special cases where it is known that the Delaunay complex has size $O(n)$. Most such cases are based on input models that guarantee the points are spaced according to some Poisson process. The analysis invariably depends on showing that the complex is everywhere locally sparse in the sense that every vertex participates in at most a constant number of simplices. It will not be hard to show that a similar bound holds for every slice and, in particular, every slice has linear size (Lemma 8). Then, we will show that the total number of simplices (the union of all slices) also has linear size (Theorem 9).

▶ **Lemma 8.** *For all $\alpha \geq 0$, every vertex in the weighted Delaunay complex D^α has at most $O\left(\left(\frac{1+\varepsilon}{\varepsilon}\right)^d\right)$ neighbors.*

Proof. Let p be any point in P. Let $Q = \{q_0, \ldots, q_{k-1}\}$ be the neighbors of p in D^α. Every edge of D^α has length at most 2α. By Lemma 3, every q_i must have a freezing time at least $\frac{\alpha}{1+\varepsilon}$ and thus, an insertion radius of at least $\frac{\varepsilon\alpha}{(1+\varepsilon)^3}$. So, the points of Q are contained in ball$(p, 2\alpha)$ and are all pairwise $\frac{\varepsilon\alpha}{(1+\varepsilon)^3}$-separated. By comparing the volumes of the k disjoint empty balls of radius $\frac{\varepsilon\alpha}{2(1+\varepsilon)^3}$ around the points of Q to the volume of the ball of radius that $\left(2 + \frac{\varepsilon}{(1+\varepsilon)^3}\right)\alpha$ that contains them, we get that $k = O\left(\left(\frac{1+\varepsilon}{\varepsilon}\right)^d\right)$. ◄

▶ **Theorem 9.** *Let P be a set of n points in $\mathbb{R}^d$ with greedy weights. Let $\varepsilon \geq 0$ be a constant. The total size of (D^α) is $O(n)$.*

Proof. The proof follows the exact pattern of previous work on sparse filtrations (see [35, 6] for more a more detailed analysis). Each simplex of the filtration is charged to the vertex with the smallest insertion radius. By a packing volume argument analogous to that in Lemma 8, we see that no vertex is charged for more than a constant number of simplices in the final filtration. Thus the total size is $O(n)$ as desired. ◄

4   Efficient Construction

The main challenge to efficiently constructing the Sparse Delaunay Filtration is to avoid constructing the entire Delaunay complex. Doing so could easily negate any efficiency gains from sparsification. In this section, we will describe an approach based on Voronoi refinement that uses extra points called Steiner points to keep the complexity of the Delaunay complex linear in the number of points. None of the Steiner points will appear in the output filtration. They serve only to fill in large gaps that could potentially create a superlinear number of Delaunay simplices.

Earlier work in approximating the persistence diagram of the distance to Euclidean points also used Steiner points [25, 33], but in that case, the Steiner points were an essential part of the filtration. Steiner points and Voronoi refinement have also been used in output-sensitive algorithms to construct D^α [36]. In that case, the output could still be superlinear in the input size depending on the arrangement of the points and the choice of α.

Flip-based algorithms for computing the Delaunay triangulation start the insertion of a new point by flipping a new vertex in with a $(1, d+1)$-flip. This requires that the new point is contained in one of the simplices of the current triangulation. If we ignore or discard simplices in the Delaunay triangulation that do not appear in the subcomplex D^α, then we cannot necessarily flip in new points, because we could have discarded a simplex containing the new point. If we maintain the full Delaunay triangulation at all times, we might store too many simplices. To balance between the two, we use Steiner point to give a sparse representation of the regions far from the input points at a given scale. In Section 4.1, we explain how these Steiner points are chosen. Then, in Section 4.2, we show why these Steiner points do not affect the output. The full algorithm is then presented in Section 4.3, and a detailed analysis is available in the full version.

4.1   Voronoi Refinement

Voronoi cells are polyhedra whose vertices we will call *corners* to distinguish them from the input vertices. We will assume that the affine closure of the points is d-dimensional, so every Voronoi cell has at least one corner. Let $P \subset \mathbb{R}^d$ and let $y \in P$ be any point. Let z be the nearest neighbor of y in P, and let x be the corner of $\text{Vor}_P(y)$ farthest from y. The *aspect* of $\text{Vor}_P(y)$ is

$$\text{aspect}(y) := \frac{\|y - x\|}{\|y - z\|}.$$

The point set P is τ-*well-spaced* if $\mathrm{aspect}(y) < \tau$ for all $y \in P$. There are many advantages to well-spaced points when constructing Delaunay triangulations. A major advantage is that the number of simplices incident to any vertex will be at most a constant. So, the total complexity of the Delaunay triangulation of n well-spaced points is at most $O(n)$.

Voronoi refinement is a variant of Delaunay refinement [10, 31] and is commonly used in mesh generation (see the books by Edelsbrunner [16] and Cheng et al. [9] for more details). The corners of a Voronoi cell are the circumcenters of their dual Delaunay simplices. The basic Voronoi refinement algorithm is to add the farthest corner of any cell $\mathrm{Vor}_P(y)$ for which $\mathrm{aspect}(y) > \tau$. Repeating this process eventually produces a τ-well-spaced set of points. Moreover, the total number of points in the output is asymptotically optimal [31, 34], i.e., the size is within a constant factor of any τ-well-spaced superset of P.

4.2 Why the Steiner points don't appear in the filtration

Every edge in D^α is induced by the intersection of two clipped Voronoi cells, so the length of every edge is at most 2α. The following lemma is the key to guarantee that the Sparse Delaunay Filtration we construct contains no Steiner points as vertices. It shows that the Steiner points are always more than 2α away from any other points, and therefore, there can be no edges incident to a Steiner point in D^α.

▶ **Lemma 10.** *Let ε and α be nonnegative real numbers. Let P and S be subsets of $\mathbb{R}^d$ such that no point of S is within 2α and no point of P is within $\varepsilon\alpha$ of any other point of $P \cup S$. If S' is formed by adding Steiner points to S at the far corners of Voronoi cells with aspect greater than $\frac{2}{\varepsilon}$, then no point of S' will be within 2α of any other point of $P \cup S'$.*

Proof. It suffices to show that the spacing condition holds after the insertion of each Steiner point x. Let y be the point whose Voronoi cell was refined by the addition of x and let z be the nearest neighbor of y. Then, $\|y - z\| \geq \varepsilon\alpha$ and the aspect of the cell is bounded as

$$\frac{2}{\varepsilon} < \frac{\|y - x\|}{\|y - z\|} \leq \frac{\|y - x\|}{\varepsilon\alpha}.$$

It follows that $\|y - x\| > 2\alpha$. Because x was in the Voronoi cell of y, it follows that the distance to any other point is also greater than 2α. ◀

4.3 The Full Algorithm

In the preprocessing phase of the algorithm, we compute a greedy permutation of P. During this computation, we also compute for each p_i, the nearest predecessor in the ordering as well as its distance, the insertion radius r_i. The radius will be used to establish the weights and define the waves. The nearest predecessor will be used for point location when inserting new points.

The algorithm then proceeds by constructing the filtration one wave at a time in order of the greedy permutation. That is, for wave w, the filtration is constructed for the interval $[(1 + \varepsilon)^w, (1 + \varepsilon)^{w+1}]$. Moreover, as the construction increases the density of points as it goes, it also decreases the radius, so the simplices of the filtration are discovered in reverse order.

At the start of each wave, some set U of points have already been inserted, and some set F of points will be inserted into to the Delaunay triangulation. The points U are unfrozen throughout the wave so their weights will always be zero. The points F are frozen at the start of the wave interval, so their weights will vary equally in time. We start the wave by locating the Delaunay simplices containing each of the points of F. Each pair of a point and the d-simplex that contains it forms a $(d + 1)$-simplex. We compute the flip time for each of these simplices and store the flip in the event queue.

Processing the flips only requires that we remove the next flip from the event queue. We check that the simplices are still present in the complex, i.e., that no other flip removed some of its subsimplices. Then, we execute the flip, updating the Delaunay triangulation, and relocating uninserted points of F that were in the removed simplices.

For each flip, we update the birth times of all simplices that may have been affected. That is, if S is the set of $d + 2$ points involved in the flip at scale α, we will process each simplex $\sigma \subseteq S$ starting with the highest dimensions. The tentative birth time of every simplex is computed assuming that the structure of the triangulation will not change. A simplex is *finalized* when we add it to the filtration. A simplex is *discarded* if we have established that it will never appear in the filtration. Discarded simplices do not need to be updated in this step. For $\sigma = S$, set $\text{birth}(S) = \alpha$ and either finalize it if $\text{radius}(S) \leq \alpha$, or discard it otherwise. For simplices σ removed by the flip, either finalize it if $\text{birth}(\sigma) \geq \alpha$ or discard it otherwise. For all other simplices, if the newly computed birth time is at most α, then update $\text{birth}(\sigma)$ and finalize σ otherwise.

At the end of each wave, we perform a Voronoi refinement step, adding Steiner points until the points are $\frac{2}{\varepsilon}$-well-spaced. That is, while any Voronoi cell has aspect greater than $\frac{2}{\varepsilon}$, we add its farthest corner. As a consequence of Lemma 10, no changes to the filtration are made at this time.

The overall running time is just the cost of computing a greedy permutation, doing an incremental Delaunay triangulation with sparse refinement, and performing a constant amount of extra work per flip. These are relatively standard analyses, so in the interest of space, they have been relegated to the full version.

5   Conclusion

We have presented a linear size Delaunay filtration for n points in $\mathbb{R}^d$ as well an efficient algorithm to compute it.

It is also relevant to note that this algorithm also can be used to compute a well-spaced set of points. That is, if one keeps the final Delaunay triangulation of the input plus the Steiner points, the result will be well-spaced. Performing a more aggressive Voronoi refinement to achieve a better spacing constant then resembles a standard Voronoi/Delaunay refinement starting from a well-spaced point set. This is substantially simpler than previous algorithms to do Sparse Voronoi Refinement [24, 29] because it obviates any need to "snap" Steiner points to nearby input points. It is not obvious whether the tradeoff between a size increase from the difference in the spacing constant offsets the improvements from simplified point location.

References

1   Franz Aurenhammer, Bert Jüttler, and Günter Paulini. Voronoi diagrams for parallel halflines and line segments in space. In Yoshio Okamoto and Takeshi Tokuyama, editors, *28th International Symposium on Algorithms and Computation (ISAAC 2017)*, volume 92 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 7:1–7:10. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2017.

2   Magnus Bakke Botnan and Gard Spreemann. Approximating persistent homology in Euclidean space through collapses. *Applicable Algebra in Engineering, Communication and Computing*, 26(1):73–101, 2015. `arXiv:1403.0533`.

3   A. Bowyer. Computing dirichlet tessellations. *The Computer Journal*, 2(24):162–166, 1981.

4   Kevin Q. Brown. Voronoi diagrams from convex hulls. *Information Processing Letters*, 9(5):223–228, 1979.

5 Mickaël Buchet, Frédéric Chazal, Steve Y. Oudot, and Donald R. Sheehy. Efficient and robust persistent homology for measures. In *ACM-SIAM Symposium on Discrete Algorithms*, pages 168–180, 2015.

6 Nicholas J. Cavanna, Mahmoodreza Jahanseir, and Donald R. Sheehy. A geometric perspective on sparse filtrations. In *Proceedings of the Canadian Conference on Computational Geometry*, 2015.

7 Frédéric Chazal, Vin de Silva, Marc Glisse, and Steve Oudot. *The Structure and Stability of Persistence Modules*. SpringerBriefs in Mathematics. Springer International Publishing, 2016.

8 Frédéric Chazal and Steve Y. Oudot. Towards persistence-based reconstruction in Euclidean spaces. In *Proceedings of the 24th ACM Symposium on Computational Geometry*, pages 232–241, 2008.

9 Siu-Wing Cheng, Tamal K. Dey, and Jonathan Richard Shewchuk. *Delaunay Mesh Generation*. CRC Press, 2012.

10 L. Paul Chew. Guaranteed-Quality Mesh Generation for Curved Surfaces. In *Proceedings of the Ninth Annual Symposium on Computational Geometry*, pages 274–280, 1993.

11 Aruni Choudhary, Michael Kerber, and Sharath Raghvendra. Improved topological approximations by digitization. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2675–2688, 2019.

12 Kenneth L. Clarkson. Nearest neighbor searching in metric spaces: Experimental results for 'sb(s)'. Preliminary version presented at ALENEX99, 2003.

13 Tamal K. Dey, Dayu Shi, and Yusu Wang. Simba: An efficient tool for approximating rips-filtration persistence via simplicial batch-collapse. In *24th Annual European Symposium on Algorithms*, pages 206:1–206:16, 2016.

14 M.E. Dyer and A.M. Frieze. A simple heuristic for the p-centre problem. *Operations Research Letters*, 3(6):285–288, 1985.

15 Herbert Edelsbrunner. The union of balls and its dual shape. *Discrete & Computational Geometry*, 13:415–440, 1995.

16 Herbert Edelsbrunner. *Geometry and Topology for Mesh Generation*. Cambridge University Press, 2001.

17 Herbert Edelsbrunner, David G. Kirkpatrick, and Raimund Seidel. On the shape of a set of points in the plane. *IEEE Transactions on Information Theory*, 29(4):551–559, 1983.

18 Herbert Edelsbrunner, David Letscher, and Afra Zomorodian. Topological persistence and simplification. *Discrete & Computational Geometry*, 4(28):511–533, 2002.

19 Herbert Edelsbrunner and Raimund Seidel. Voronoi diagrams and arrangements. *Discrete & Computational Geometry*, 1(1):25–44, 1986.

20 Herbert Edelsbrunner and Nimish R. Shah. Incremental topological flipping works for regular triangulations. *Algorithmica*, 15, 1996.

21 Teofilo F. Gonzalez. Clustering to minimize the maximum intercluster distance. *Theor. Comput. Sci.*, 38:293–306, 1985.

22 Leonidas J Guibas. Kinetic data structures – a state of the art report. In *Proc. Workshop Algorithmic Found. Robot*, pages 191–209, 1998.

23 Sariel Har-Peled and Manor Mendel. Fast construction of nets in low dimensional metrics, and their applications. *SIAM Journal on Computing*, 35(5):1148–1184, 2006.

24 Benoît Hudson, Gary Miller, and Todd Phillips. Sparse Voronoi Refinement. In *Proceedings of the 15th International Meshing Roundtable*, pages 339–356, Birmingham, Alabama, 2006. Long version available as Carnegie Mellon University Technical Report CMU-CS-06-132.

25 Benoît Hudson, Gary L. Miller, Steve Y. Oudot, and Donald R. Sheehy. Topological inference via meshing. In *Proceedings of the 26th ACM Symposium on Computational Geometry*, pages 277–286, 2010.

26 Barry Joe. Three-dimensional triangulations from local transformations. *SIAM J. Sci. Stat. Comput.*, 10:718–741, 1989.

27   C. L. Lawson. Software for C1 surface interpolation. In J. R. Rice, editor, *Mathematical Software*, volume III, pages 161–194. Academic, New York, 1977.

28   Charles L. Lawson. Transforming triangulations. *Discrete Mathematics*, 3:365–372, 1972.

29   Gary L. Miller, Todd Phillips, and Donald R. Sheehy. Beating the spread: Time-optimal point meshing. In *Proceedings of the 26th ACM Symposium on Computational Geometry*, pages 321–330, 2011.

30   Gary L. Miller and Donald R. Sheehy. A new approach to output-sensitive construction of voronoi diagrams and delaunay triangulations. *Discrete & Computational Geometry*, 52(3):476–491, 2014. `doi:10.1007/s00454-014-9629-y`.

31   Jim Ruppert. A Delaunay refinement algorithm for quality 2-dimensional mesh generation. *J. Algorithms*, 18(3):548–585, 1995.

32   Raimund Seidel. On the number of faces in higher-dimensional Voronoi diagrams. In *Proceedings of the 3rd Annual Symposium on Computational Geometry*, pages 181–185, 1987.

33   Donald R. Sheehy. *Mesh Generation and Geometric Persistent Homology*. PhD thesis, Carnegie Mellon University, 2011. CMU CS Tech Report CMU-CS-11-121.

34   Donald R. Sheehy. New Bounds on the Size of Optimal Meshes. *Computer Graphics Forum*, 31(5):1627–1635, 2012.

35   Donald R. Sheehy. Linear-size approximations to the Vietoris-Rips filtration. *Discrete & Computational Geometry*, 49(4):778–796, 2013.

36   Donald R. Sheehy. An output-sensitive algorithm for computing weighted α-complexes. In *Proceedings of the Canadian Conference on Computational Geometry*, 2015.

37   D. F. Watson. Computing the n-dimensional delaunay tessellation with application to voronoi polytopes. *The Computer Journal*, 24(2):167–172, 1981.