A Randomized Algorithm for Closest-Point Queries*

Kenneth L. Clarkson†

Abstract

An algorithm for closest-point queries is given. The problem is this: given a set S of n points in d-dimensional space, build a data structure so that given an arbitrary query point p, a closest point in S to p can be found quickly. The measure of distance is the Euclidean norm. This is sometimes called the *post-office problem*. The new data structure will be termed an *RPO tree*, from Randomized Post Office. The expected time required to build an RPO tree is $O(n^{\lceil d/2 \rceil(1+\epsilon)})$, for any fixed $\epsilon > 0$, and a query can be answered in $O(\log n)$ worst-case time. An RPO tree requires $O(n^{\lceil d/2 \rceil(1+\epsilon)})$ space in the worst case. The constant factors in these bounds depend on d and ϵ. The bounds are average-case due to the randomization employed by the algorithm, and hold for any set of input points. This result approaches the $\Omega(n^{\lceil d/2 \rceil})$ worst-case time required for any algorithm that constructs the Voronoi diagram of the input points, and is a considerable improvement over previous bounds for $d > 3$. The main step of the construction algorithm is the determination of the Voronoi diagram of a random sample of the sites, and the triangulation of that diagram.

1 Introduction

The post-office problem is a fundamental problem of computational geometry, having many applications in statistics, operations research, interactive graphics, coding theory, and other areas.

Several algorithms that are asymptotically fast in the worst-case sense are known for this problem in the planar ($d = 2$) case. They involve the construction of the Voronoi diagram of the sites [12], [26], and the use of fast methods for searching planar subdivisions resulting from that diagram [19], [17], [11]. By these methods, a data structure requiring $O(n)$ space can be constructed in $O(n \log n)$ time, so that a query can be answered in $O(\log n)$ time.

*Received by the editors June 26, 1986; accepted for publication (in revised form) July 27, 1987.

†AT&T Bell Laboratories, Murray Hill, New Jersey 07974.

The higher-dimensional cases are much less examined and understood. Dobkin and Lipton have described a data structure requiring $O(n^{2^{d+1}})$ time and space to construct, giving a query time of $O(\log n)$ [10]. Chazelle has given an algorithm for the case $d = 3$ that requires $O(n^2)$ preprocessing for $O(\log^2 n)$ query time [4].

Although the time and space bounds for RPO trees are rather large for large d, they are a considerable improvement over previous general bounds. The key step in the construction of the data structure is the determination of Voronoi diagrams of small subsets of the sites. (For convenience, points in S will be called *sites*.) The bounds depend on the complexity of these Voronoi diagrams. If the diagrams have $O(n)$ vertices, the construction requires an expected time bounded by $O(n^C)$, where C is a constant independent of d. Indeed, when a set of sites is uniformly distributed in a hypercube [1], or spatially Poisson-distributed [13], their Voronoi diagram has linear complexity on the average. These facts suggest that RPO trees may do considerably better in practice than the worst-case bounds would show. On the other hand, in the worst case Voronoi diagrams may require $\Omega(n^{\lceil d/2 \rceil})$ storage [18], [24], so in a sense any algorithm using Voronoi diagrams could not perform too much better than RPO trees.

1.1 Overview

The initial observation for the data structure is just this: if we want to find a closest site in S to a point p, then knowing a closest site to p in some $R \subset S$ can help restrict the search in S. The terms *candidate sets* and *candidate sites* help to formalize this notion.

DEFINITION. For a given subset $R \subset S$, and any point x, let r_x denote the distance of x to a closest site in R. Then the *candidate region* $C(x)$ for x, relative to R, is the ball of points whose distance to x is less than r_x. The corresponding closed ball will be denoted $\overline{C}(x)$. The *candidate set* for x is $S \cap C(x)$. The candidate region for a set of points A is $C(A) = \cup_{x \in A} C(x)$, with the candidate set $S \cap C(A)$.

Thus, for a query point p and region A with $p \in A$, the set $S \cap \overline{C}(A)$ contains all the closest sites to p. If q is a closest site in R to p, then the candidate set $S \cap C(A)$ contains all sites closer to p than q. The key idea is to find some $R \subset S$, and some collection of regions, such that for every region A in the collection, the candidate set of A relative to R contains few sites.

Such a collection of regions can be found using random sampling, as follows: take a random sample R of the sites, determine the Voronoi diagram $\mathcal{V}(R)$ of that sample, and then compute $\Delta(\mathcal{V}(R))$, a triangulation of the Voronoi diagram. (Voronoi diagrams are defined in §2; triangulations are discussed in §3.) The result is a collection of simple regions with the following properties:

- The union of the regions covers $\Re^d$, that is, $\Re^d = \cup_{A \in \Delta(\mathcal{V}(R))} A$;
- The number of regions is $O(r^{\lceil d/2 \rceil})$, for $r \to \infty$, where r is the size of R;

- With high probability, the candidate sets $S \cap C(A)$ are "small" for all regions $A \in \Delta(\mathcal{V}(R))$, specifically, $|S \cap C(A)| = nO(\log r/r)$ as $r \to \infty$;
- The regions in $\Delta(\mathcal{V}(R))$ are simple, so that for point p and $A \in \Delta(\mathcal{V}(R))$, we can tell in $O(1)$ time if $p \in A$, for fixed dimension d;
- For each $A \in \Delta(\mathcal{V}(R))$, there is a site $q \in R$ such that all points in A are as close to q as to any other site in R.

These properties suggest a two-step process for answering closest-point queries: given query point p, determine a region $A \in \Delta(\mathcal{V}(R))$ that contains it, then determine the closest site to p in $R \cup (S \cap C(A))$ by linear search. For a suitable sample size, with high probability this procedure is faster than directly searching S. By repeatedly taking random samples until a sample is found for which the corresponding candidate sets are all small, a data structure with an improved worst-case query time can be constructed. Since a random sample will satisfy this condition with high probability, on average only $O(1)$ sampling repetitions need be done.

Rather than search the candidate sets in linear time, this construction can be applied recursively, using a sample size r that is independent of the number of sites. The resulting search structure is an RPO tree, in which the number of children of a node is independent of the number of sites, as is the size of the set of sites associated with each leaf node.

Each node t of an RPO tree corresponds to a collection of sites S' that contains the closest site to a set of potential query points. If t is an internal node, a suitable sample $R' \subset S'$ is found, and for each $A \in \Delta(\mathcal{V}(R'))$, there is a child t' of t for which a record t'.region is A. The children of t form a list t.children. A closest-point query can be answered by tracing down from the root, moving from a current node t to a child $t' \in t$.children, whose associated t'.region contains the query point. If t is a leaf node, the sites *t.sites* associated with t are given a linear search to answer the query.

The procedures *Make_RPO_Tree* and *Answer_Query* are shown in Fig. 1. The procedure *New_RPO_Tree* returns a new RPO tree, whose regions and subtrees are subsequently defined. From Theorem 16, the sample size r should be at least about $(d+1)^3$. The constant K should be no smaller than r. The constant $\alpha_{r,d} = O(\log r/r)$ is defined in Theorem 16.

Before making a more detailed description of the algorithm, it may be helpful to consider informally the simplest interesting example of a set $S \cap \overline{C}(A)$, which occurs when A is a triangular region in the plane, a region in the triangulation of the Voronoi diagram of a sample R. In Fig. 2, the set $R = \{q, p_1, \ldots, p_5\}$, and A has vertices a, b, and c, part of a triangulation of the Voronoi region $\mathcal{V}_q$. As will be shown in §4, the region $\overline{C}(A)$ has a particularly simple description: it is simply $\overline{C}(a) \cup \overline{C}(b) \cup \overline{C}(c)$. Since a, b, and c are vertices of $\mathcal{V}_q$, the circles bounding $\overline{C}(a)$, $\overline{C}(b)$, and $\overline{C}(c)$ are *Delaunay circles* of the Voronoi diagram $\mathcal{V}_R$. To restate this fact, suppose $p \in A$. Then since $q \in S$, the closest site to p is

```
function Make_RPO_Tree(S : Set_of_Sites) return t : RPO_Tree;

t←New_RPO_Tree;
if |S| < K then t.leaf←true; t.sites←S;
else
    t.leaf←false;
    repeat choose random sample R ⊂ S until ∀A ∈ Δ(𝒱(R)), |S ∩ C(A)| ≤ α_{r,d}|S|;
    t.children←∅;
    for A ∈ Δ(𝒱(R)) do
        t'←Make_RPO_Tree(S ∩ C(A));
        t'.region←A; t'.site← site q such that A ⊆ V_q; t.children←t.children ∪ {t'};
    od;
fi;
end function Make_RPO_Tree;

function Answer_Query(t : RPO_Tree; p : query_point) return closest : site;
current_closest← any site in R;
while not t.leaf do
        choose any t' ∈ t.children with p ∈ t'.region;
        if t'.site closer to p than current_closest then current_closest←t'.site;
        t ← t';
od;
closest← site closest to p among those in t.sites ∪ {current_closest};
end function Answer_Query;
```

Figure 1: Procedures Make_RPO_Tree and Answer_Query.

contained in the disk defined by the circle centered at p that passes through q. This disk is contained in the union of the Delaunay disks at a, b, and c.

For the RPO construction to work, with high probability all these Delaunay disks should contain few sites. Why should this be? The reason is based on the fundamental fact that these Delaunay disks contain no sample sites. Intuitively, this provides some evidence that these disks contain few sites: if some arbitrary disk contains a large fraction of the sites, then with high probability, some sample site will be chosen from that disk, and it cannot be a Delaunay disk. This argument is made precise in §4.3.

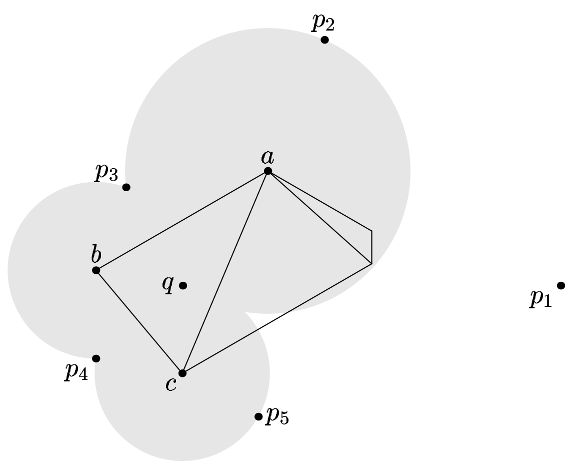

Figure 2: A triangle $A \in \Delta(\mathcal{V}_q)$ and $\overline{C}(A)$.

1.2 Outline of the paper

To complete the description of the algorithms, it is necessary to specify the triangulation procedure Δ, and to characterize formally the candidate regions $C(A)$ for $A \in \Delta(\mathcal{V}(R))$. To analyze the algorithms, we must bound the number of children that a node can have, that is, the number of regions in $\Delta(\mathcal{V}(R))$, and also bound the size of the resulting subproblems, that is, the size of each $C(A) \cap S$. Before addressing these questions, some notation and basic lemmas will be given in §2. Many readers should be able to skim most of this section, or refer to it as needed. In §3, the triangulation procedure is given, and a bound on the size of its output is developed. In §4, it is shown that the candidate regions to be used have simple descriptions, generalizing the above example. It is also shown that with high probability, all the corresponding candidate sets have few sites. Also given in this section are the modifications to the algorithms for handling a variant of the post-office problem, in which all closest sites to a query point are desired. In §5, the complexity analysis of the algorithms is completed. Some concluding remarks are made in §6, with discussion of subsequent and related work.

The basic idea for the RPO data structure is simple, and the critical algorithmic step is the fundamental operation of computing the Voronoi diagram, followed by triangulation. Nonetheless, several factors complicate the discussion. The algorithms generalize for an arbitrary dimension, so that the descriptions and proofs of correctness are abstract. An operation of triangulation must be applied to the unbounded polyhedral sets of a Voronoi diagram, as must the determination of candidate regions. This is best done using the notion of "points at infinity," considering an unbounded polyhedral set as the convex hull of a set of points, some of which are at infinity. This idea is made precise using *two-sided space*, described in the next section. The sample R may be degenerate, that is, not have full affine dimension. This possibility must be accounted for. These factors imply that the description must be more abstract and complicated than it otherwise would be.

2 Notation, terminology, and background

The notation in this paper will follow [14] in general, and use basic results from that text. The concepts of *oriented projective geometry* [27] will also play a large role. The following notation is gathered here for reference:

$\Re^d$ denotes d-dimensional Euclidean space;

$A + B$ is the pointwise sum $\{x + y \mid x \in A, y \in B\}$, for $A, B \subset \Re^d$;

$x + A$ and $A + x$ denote $\{x\} + A$, for $x \in \Re^d$;

αA denotes the product $\{\alpha x \mid x \in A\}$, for a real number α and $A \subset \Re^d$;

$S_{x,y}$ denotes the sphere that has center x and that contains y. $B_{x,y}$ and $\overline{B}_{x,y}$ denote the corresponding open and closed balls;

A *flat* $F \subset \Re^d$ is an *affinely closed* set, that is, if $x, y \in F$, then the straight line through x and y is contained in F.

$\operatorname{aff} A$ denotes the affine closure of a point set $A \subset \Re^d$, that is, the intersection of all flats containing A;

$\dim A$ denotes the affine dimension of A, that is, the dimension of the linear subspace $(\operatorname{aff} A) - p$, for $p \in A$. A k-flat F has $k = \dim F$;

$\operatorname{conv} A$ denotes the convex closure of A, that is, the intersection of all convex sets containing A;

$\operatorname{relint} A$ is the interior of A relative to its affine closure;

$\operatorname{relbd} A$ is the boundary of A relative to its affine closure.

Rays and cones. For $x, y \in \Re^d$, let $\operatorname{ray}_x y$ denote

$$\{x + \alpha(y - x) \mid \alpha \geq 0\}.$$

A *cone* C with apex a is a subset of $\Re^d$ such that $\operatorname{ray}_a y \subset C$ if and only if $y \in C$, for $y \in \Re^d$.

Polyhedral sets and polytopes. A *polyhedral set* is the intersection of a finite number of closed halfspaces, and a *polytope* is a bounded polyhedral set. A polyhedral cone is a cone that is a polyhedral set. A d-polytope (d-polyhedral set) P satisfies $d = \dim P$.

A *supporting hyperplane* h of a polyhedral set P satisfies $h \cap P \neq \emptyset$ and $h^+ \cap P = \emptyset$, where h^+ is an open halfspace defined by h. A *face* of a polyhedral set P is the intersection of P with a supporting halfspace. Vertices, edges, and facets are faces of affine dimension 0, 1, and $d-1$, respectively, for a d-polyhedral set. In general, a face of P with dimension k is a *k-face*, and the set of such faces is $f_k(P)$. The set $f_0(P)$ of vertices (or *extreme points*) of a polyhedral set P will be denoted by $\operatorname{vert} P$.

Two polyhedral sets A and B are said to be *combinatorially equivalent* if there is a bijective mapping Λ from the faces of A to those of B such that $F \subset G$ if and only if $\Lambda(F) \subset \Lambda(G)$, for all F and G faces of A.

The set of *extreme rays* of a polyhedral set P is the set of rays e emanating from the origin such that there is some point q for which $q+e$ is an edge (1-face) of P. The convex hull of the extreme rays of P is the *characteristic cone* $\operatorname{cc} P$. From [14, 2.5.2], if $x, y \in P$ and e is a ray from the origin, then $x + e \subseteq P$ if and only if $y + e \subseteq P$. (Looking ahead, this is equivalent to the condition that e corresponds to a "point at infinity" in P, and that P is convex even when such points are included.) This fact and the convexity of P imply that $x + \operatorname{cc} P \subset P$ for any $x \in P$. It can be shown that $\operatorname{cc} P$ is the maximal such cone.

A set is said to be *line-free* if it contains no straight lines (no 1-flats). A line-free cone is pointed [14, p. 24], that is, it has only one apex, which is a vertex. Many basic facts about polytopes generalize nicely to line-free polyhedral sets, using the notion of "ideal points" defined below.

Complexes. A *complex* is a collection of polyhedral sets such that every face of a polyhedral set in the complex is also in the complex, and the intersection of two polyhedral sets in the complex is a face of each of them. (In the complexes considered here the empty set $\emptyset$ is a face.) A polyhedral set of dimension k in a complex is a *k-face* of that complex, and the terminology of vertices, edges, and facets carries over for complexes. The *facial lattice* of a complex is the set of faces of the complex, together with the inclusion relations between those faces.

One example of a complex is the *boundary complex* $\mathcal{B}(P)$ of a polyhedral set P, the set of facets of P and their faces. Another example of a complex is the Voronoi diagram of a set of sites, described below.

Two-sided space, homogeneous coordinates, and ideal points. It will be helpful conceptually and computationally to use the notion of "points at infinity," also known as *ideal* points, as opposed to the usual *real* points in $\Re^d$. These classes of points together make up what will be denoted T^d, or *two-sided space*. (The name will be explained below.)

To represent points in T^d, *homogeneous* coordinates will be used: a real point $x \in \Re^d$ is represented by $x_h = [x_r; x_s]$ if $x = x_r/x_s$, where $x_r \in \Re^d$ and $x_s \in \mathcal{R}$, $x_s > 0$. (The terminology is borrowed from projective geometry [23], although in this case, the coordinates cannot really be said to be homogeneous.) An ideal point $x \in T^d$ is represented by the homogeneous coordinates $[x_r; 0]$, where $x_r \in \Re^d$ and $x_r \neq 0$. The point x can be considered the "endpoint" of $\operatorname{ray}_0 x_r$. If x is an ideal point and y is real, we will say that $\operatorname{conv}\{x, y\}$ is $\operatorname{ray}_y(x_r + y)$. Indeed, if $z \in \operatorname{conv}\{x, y\}$, for any points x and y, then for any representations z_h, x_h, y_h, we have $z_h = \alpha_x x_h + \alpha_y y_h$, for some $\alpha_x, \alpha_y \geq 0$, and conversely. This provides a general definition of convex combination for points in T^d.

Note that homogeneous coordinate representations are not unique: if x_h is a homogeneous representation of x, then so is βx_h, for any $\beta > 0$. (This convention is different from that of projective geometry, where β need only be nonzero. This follows [21] and [15], and is needed to distinguish ideal points in "opposite directions.") The two-sided nature of T^d derives from its containment of two copies of $\Re^d$, since $[x_r; 1]$ and $[x_r; -1]$ represent distinct points in T^d for every $x_r \in \Re^d$. There is a correspondence between points in T^d and the d-sphere

$$S^d = \{x \in E^{d+1} \mid \|x\| = 1\}.$$

A point with homogeneous coordinates x_h corresponds to $x_h/\|x_h\|$, where x_h is interpreted as a point in E^{d+1}. The ideal points of T^d correspond to those points of S^d on the hyperplane $x_s = x_{d+1} = 0$. The two halves of S^d separated by the set of ideal points correspond to the two sides of T^d.

In general, the only points in T^d considered will be those satisfying $x_s \geq 0$. This set of points is termed the "front range" of T^d, and will be denoted by F^d. A closed convex set $P \subset E^d$ will be extended by including $[b - a; 0]$ in P whenever $\text{ray}_a\, b \subset P$. This implies that the set of points on S^d corresponding to P is also closed. A straight line will thus have two ideal "endpoints," and so on for all flats. (Note that this gives a meaning to "flat" that is different from Stolfi's [27].) This convention will extend the definition of the sum $A + B$ for unbounded A and B. The notation for a sphere $S_{x,y}$ can be extended to allow the center x to be an ideal point. In this case, $S_{x,y}$ is a hyperplane normal to x_r and passing through y. The closed ball $\overline{B}_{x,y}$ is the corresponding closed halfspace. An analytic relation describing such spheres is given below in the discussion of Voronoi diagrams.

The set $\text{re}\, A$ will denote the real points of A, and $\text{id}\, A$ will denote the ideal points of A. Note that $\text{re}\, A = \text{re}\, B$ implies that $\text{id}\, A = \text{id}\, B$, for A and B closed and convex.

With this extension to the front range F^d, a line-free unbounded polyhedral set $P \subset \Re^d$ has an implicit additional defining halfspace. That is, $\text{id}\, P$ is an additional facet of P, the *ideal facet*. This facet corresponds to $\text{cc}\, P$ in a natural way, and the correspondence extends to the faces of $\text{id}\, P$, so that the vertices of $\text{id}\, P$ correspond to the extreme rays of $\text{cc}\, P$. Thus $\text{vert}\, P$ is extended to include ideal points. The following simple lemma helps in generalizing facts about polytopes to facts about polyhedral sets.

Lemma 1 *Let C be a polyhedral pointed cone with apex a. Let h be a supporting hyperplane of C with $h \cap C = \{a\}$. Let ν be a normal vector to h contained in the same halfspace containing C. Then $P = C \cap (h + \nu)$ is a polytope.*

Proof. See Appendix A. □

It is easy to show that $C = \cup_{y \in A} \text{ray}_x\, y$, and that $x \in P$ if and only if $x' = [x - a; 0] \in \text{id}\, C$. This bijective map satisfies $(\alpha x + \beta y)' = \alpha x' + \beta y'$, for $\alpha + \beta = 1$. This implies that $\text{id}\, C$ and P are combinatorially equivalent. Thus Lemma 1 brings $\text{id}\, C$ into the "real" world of $\Re^d$.

The following lemma is a generalization of [14, 2.4.5] from polytopes to line-free polyhedral sets.

Lemma 2 *If $P \subset E^d$ is a line-free d-polyhedral set then $P = \operatorname{conv} \operatorname{vert} P$.*

Proof. See Appendix A. □

Simplices and triangulations. A *simplex* is a simple kind of polyhedral set: a d-simplex is a polyhedral set with $d+1$ vertices and affine dimension d. Note that vertices will be allowed to be ideal. For example, a triangle with two ideal vertices is a cone, and a triangle with one ideal vertex is bounded by a line segment and by two parallel rays from the endpoints of that line segment.

A *simplicial complex* is a complex composed of simplices. A *triangulation* $\mathcal{T}$ of a complex $\mathcal{C}$ is a simplicial complex that is a subdivision of $\mathcal{C}$. Every vertex of $\mathcal{T}$ is a vertex of $\mathcal{C}$, every facet of $\mathcal{T}$ is a simplex, and the union of the facets of $\mathcal{T}$ is the union of the facets of $\mathcal{C}$.

In §3, a particular kind of triangulation of $\mathcal{C}$, denoted $\Delta(\mathcal{C})$, will be described. Construction of this complex is an essential procedure in the algorithm given in this paper. The complexity of $\Delta(\mathcal{C})$ and of its construction procedure are also considered in §3.

The triangulation Δ may involve "simplices" of even greater generality than those shown above. For example, suppose $\dim S = 2$ but $\dim R = 1$, specifically, R is a set of sites on the x axis. Then each Voronoi region $\mathcal{V}_q$ of R will be a strip bounded by two parallel lines. This can be viewed as an interval A, that is, a 1-simplex, added to the y-axis. That is, $\mathcal{V}_q = A + l$, where l is the y-axis line. In general, when $d > \dim R$, the regions of $\Delta(\mathcal{V}(R))$ will have the form $A + f$, where A is a simplex and f is a flat orthogonal to $\operatorname{aff} A$. This generalization is formalized by Lemma 6.

Duality. For a complex $\mathcal{C}$, a *dual* $\mathcal{D}$ of $\mathcal{C}$ is another complex for which there is an inclusion-reversing correspondence between faces of $\mathcal{C}$ and those of $\mathcal{D}$. That is, there is a bijective mapping Ψ from the set of faces of $\mathcal{C}$ to those of $\mathcal{D}$ such that for faces F and G of $\mathcal{C}$, the inclusion $F \subset G$ holds if and only if $\Psi(F) \supset \Psi(G)$. The facial lattice of $\mathcal{C}$ can be determined from the facial lattice of $\mathcal{D}$, and vice versa.

One particular dual relation between polyhedral sets is quite useful. For a set $A \subset E^d$, the polar set A^* is

$$\{y \mid y_h \cdot x_h \geq 0 \text{ for all } x \in A\}.$$

When P is a line-free d-polyhedral set in E^d with the origin in its interior, the polar set P^* of P is a polytope, even when P is unbounded. Moreover, P^* is dual to P. The ideal facet of P, if any, corresponds to the origin, which will be a vertex of P^*, for unbounded P.

Voronoi diagrams. The *Voronoi diagram* $\mathcal{V}(R)$ of a set of sites (points) R in $\Re^d$ is the partition of $\Re^d$ into blocks, such that all points in a block have exactly the same closest sites. The Voronoi region $\mathcal{V}_q$ associated with a site $q \in R$ is the polyhedral set containing all points at least as close to q as they are to any other site. If $v \in \text{re vert}\, \mathcal{V}_q$ for some site q, then the sphere (ball) centered at v with radius $\|v - q\|$ is termed a *Delaunay* sphere (ball) of the sites R. (A ball in the plane is also called a disk.) At least $d + 1$ sites are on a Delaunay sphere, and none are inside it.

It is well known that $\mathcal{V}_q$ is unbounded if and only if q is on the convex hull of R. Furthermore, each unbounded edge of $\mathcal{V}_q$ is normal to a facet of $\text{conv}\, R$ that contains q.

Brown [3] has shown that the computation of a Voronoi diagram in $\Re^d$ can be reduced to the problem of computing a convex hull in E^{d+1}. The reduction is done by means of a mapping from E^d to E^{d+1}. One mapping that achieves this reduction is the function $\Upsilon : E^d \mapsto E^{d+1}$, which sends $y \in E^d$ to the point $(y_1, \ldots, y_d, -y \cdot y/2) \in E^{d+1}$. For $x, q \in E^d$, we have $y \in S_{x,q}$ if and only if

$$(x_1, \ldots, x_d, 1) \cdot \Upsilon(y) = (x_1, \ldots, x_d, 1) \cdot \Upsilon(q).$$

This implies $\text{aff}\, \Upsilon(S_{x,q})$ is a hyperplane. Furthermore, if $\overline{B}_{x,q}$ is a Delaunay ball, then $\Upsilon(R) \cap \Upsilon(\overline{B}_{x,q})$ is empty, so that $\text{aff}\, \Upsilon(S_{x,q})$ contains a facet of $\text{conv}\, \Upsilon(R)$. It follows that $\text{conv}\, \Upsilon(R)$ gives a dual complex to $\mathcal{V}(R)$.

Note also that the analytic condition for $y \in S_{x,q}$ can be extended coherently to ideal x by $x_h \cdot (\Upsilon(y) - \Upsilon(q)) = 0$, where x_h is interpreted as a point in E^{d+1}. For ideal y and x, the appropriate condition is $y_r \cdot x_r = 0$.

3 Triangulating polytopes and Voronoi regions

3.1 A triangulation procedure and its correctness The procedure Δ to be used for triangulating a complex $\mathcal{C}$ is straightforward: the procedure produces a set of simplices triangulating each face of $\mathcal{C}$, considering these faces in increasing order of their affine dimension. Note that if $1 \geq \dim P$, then the face P is a simplex. If $1 < \dim P$, then arbitrarily pick $v \in \text{vert}\, P$, and let $\Delta(P)$ be the collection

$$\{\text{conv}(\{v\} \cup S) \mid S \in \Delta(F), F \text{ a facet of } P, v \notin F\}.$$

For example, if P is a polygon, then for every edge e of P not containing vertex v, the triangle defined by e and v is in $\Delta(P)$. Note that once $\Delta(P)$ is computed, it is "fixed," so that the same triangulation of P is used whenever a face is triangulated for which P is a facet.

To apply Δ to a polyhedral set, that set must have a vertex. However, not all polyhedral sets have vertices. (An example of such a polyhedral set is given in the discussion of triangulations in §2.) We will first show that Δ can be applied to line-free polyhedral sets, and then discuss the extension of Δ to arbitrary polyhedral sets.

Lemma 3 *A line-free polyhedral set P has a vertex.*

Proof. If P is a real polyhedral set, the result is a special case of [14, 2.4.6]. If P is an ideal polyhedral set (say some ideal facet of an unbounded polyhedral set), then the result follows from Lemma 1 and [14, 2.4.6]. □

This lemma implies that some choice $v \in \operatorname{vert} F$ can be made in Δ. To show that P is the union of the simplices in $\Delta(P)$, the following lemma is useful.

Lemma 4 *For a line-free polyhedral region P, if $v \in \operatorname{vert} P$ and $a \in P$, then the point $b = \operatorname{ray}_v a \cap \operatorname{relbd} P$ is on a facet of P that does not contain v.*

Proof. See Appendix A. □

Theorem 5 *Given a complex $\mathcal{C}$ containing only line-free polyhedral sets, $\Delta(\mathcal{C})$ is a triangulation of that complex. (Note that the simplices in $\Delta(\mathcal{C})$ may have ideal vertices.)*

Proof. Induction on dimension will be applied to each polyhedral set P in $\mathcal{C}$ to show that $\Delta(P)$ returns a set of simplices covering P. Let $a \in P$. Then $\operatorname{ray}_v a$ intersects $\operatorname{relbd} P$ at v and at some point b. (If P is ideal, map the points involved to the polytope of Lemma 1.) By the previous lemma, b is on some facet F not containing v. Since by inductive assumption, b is in a simplex A of a triangulation of F, it follows that $a \in \operatorname{conv}(\{v\} \cup A)$, and the set of regions returned by Δ covers P.

The other properties of a triangulation follow by similar straightforward induction. □

To extend the triangulation procedure Δ to polyhedral sets that do not have vertices, the following lemma is useful.

Lemma 6 *Let $P \subset T^d$ be a closed polyhedral set. Then there is a subspace L of maximum affine dimension for which $x + L \subset P$ for any $x \in P$. Furthermore, if L^* is any flat that is orthogonally complementary to L, then $P = (P \cap L^*) + L$, where $P \cap L^*$ is a line-free polyhedral set.*

Proof. (Recall that L^* and L orthogonal means that L and the subspace $L^* - x$ are orthogonal, where $x \in L^*$. That is, every vector in $L^* - x$ is perpendicular to every vector in L. Since L^* and L are extended to include ideal points, the sum $L + L^*$ is the front range of T^d.)

For $\operatorname{re} P$, the lemma is a restatement of [14, 2.5.4]. The extension to $\operatorname{id} P$ follows from this, since $\operatorname{re} P = \operatorname{re}(P \cap L^*) + \operatorname{re} L$ implies equality for the ideal parts as well. □

In the particular case where P is a Voronoi region, the flat L^* can be taken to be $\operatorname{aff} R$, so that L is $(\operatorname{aff} R)^\perp$, the subspace orthogonal to $\operatorname{aff} R$. Observe that if l is a straight line contained in a Voronoi region V_q, then $l = (\operatorname{conv}\{a, c\}) \cup (\operatorname{conv}\{a, -c\})$, where $a \in \operatorname{re} l$ and $c \in \operatorname{id} l$. For any other site $q' \in R$, $c \in V_q$ implies that $q' \notin B_{c,q}$, so that $c_h \cdot (\Upsilon(q') - \Upsilon(q)) \geq 0$. But $-c \in V_q$ as well, so $c_h \cdot (\Upsilon(q') - \Upsilon(q)) \leq 0$. Therefore for any $q' \in R$, we have $q' \in S_{c,q}$, and so $\operatorname{aff} R \subseteq S_{c,q}$. That is, $\operatorname{aff} R$ is perpendicular to c, and $l \subseteq (\operatorname{aff} R)^\perp$.

To use this lemma to extend Δ for Voronoi regions containing lines, simply define $\Delta(V_q)$ to be the set of regions $\{A + (\operatorname{aff} R)^\perp \mid A \in \Delta(V_q \cap \operatorname{aff} R)\}$.

3.2 Complexity of the complex $\Delta(\mathcal{V}(R))$

In the worst case, the Voronoi diagram $\mathcal{V}(R)$ has $\Theta(r^{\lceil d/2 \rceil})$ faces. This follows from the correspondence discussed in §2 between $\mathcal{V}(R)$ and $\operatorname{conv} \Upsilon(R) \subset E^{d+1}$, and from the Upper Bound Theorem [20] applied to $(d+1)$-polytopes. As is shown below, the number of simplices in $\Delta(\mathcal{V}(R))$ has the same $\Theta(r^{\lceil d/2 \rceil})$ bound. First a bound will be proven for $\Delta(P)$, in the case where P is a *simple* polytope, defined below. Next it will be shown that for every polytope P, there is a simple polytope $\hat{P}$ with the same number of facets, such that $\Delta(\hat{P})$ has at least as many simplices as $\Delta(P)$.

Lemma 7 *For a simple d-polytope P with n facets, the triangulation $\Delta(P)$ has $O(n^{\lfloor d/2 \rfloor})$ simplices, as $n \to \infty$.*

Proof. A simple polytope satisfies the condition that every vertex of the polytope is contained in exactly d facets of that polytope. Suppose P is a simple polytope. Then the dual P^* has the property that every facet of P^* contains exactly d vertices. (The dual and some of its properties are described in §2.) That is, the facets of P^* are all simplices, which is the definition of a *simplicial* polytope. The faces of a simplex are all simplices [14, §4.1]. That is, each k-face of P^* is a k-simplex and has $k+1$ facets. This means that dually, every $(d-1-k)$-face of P is a facet of $k+1$ faces of P. Put another way, every k-face of P is a facet of $d-k$ faces of P.

What does this fact imply for $\Delta(P)$? For a polytope F, let $|\Delta(F)|$ denote the number of simplices in $\Delta(F)$. Then the definition of $\Delta(F)$ and the above fact imply that

$$\sum_{F \in f_k(P)} |\Delta(F)| = \sum_{\substack{F \in f_k(P) \\ F' \in f_{k-1}(F)}} |\Delta(F')| \leq (d-k+1) \sum_{F' \in f_{k-1}(P)} |\Delta(F')|.$$

The second relation holds because $|\Delta(F')|$ appears $d-k+1$ times in the second sum. Putting these relations together,

$$|\Delta(P)| = \sum_{F \in f_d(P)} |\Delta(F)| \leq d! \sum_{F \in f_0(P)} |\Delta(F)| = d! |\operatorname{vert} P|.$$

By the results of [20], $|\operatorname{vert} P| = O(n^{\lfloor d/2 \rfloor})$ when P is a d-polytope with n facets, so the lemma follows. □

Before proving the corresponding lemma for nonsimple polytopes, a definition is needed.

DEFINITION. Let $P \subset \Re^d$ be a d-polytope and let h be a hyperplane with $F = h \cap P$ a facet of P. Let $\hat{h}$ be a hyperplane with nonempty intersection with P, and with all vertices of $\operatorname{vert} P \setminus \operatorname{vert} F$ in the open halfspace $\hat{h}^+$. If F is in the open halfspace $\hat{h}^-$, then the polytope $\hat{P} = P \cap \hat{h}^+$ will be said to be obtained from P by *pushing* the facet F.

This operation of pushing a facet of a polytope is the dual of the operation of *pulling* a vertex [14, §5.2]. The (polar) dual polytope of a pushed polytope $\hat{P}$ can be obtained by pulling a vertex of P^*. In [14, §5.2] it is shown that the operation of pulling vertices, when applied successively to every vertex of a polytope, results in a simplicial polytope. Dually, the operation of pushing facets yields a simple polytope.

The following lemma is a restatement of [14, 5.2.2], together with some relevant discussion in that section.

Lemma 8 *Suppose $\hat{P} \subset \Re^d$ is a d-polytope obtained from the d-polytope P by pulling $v \in \operatorname{vert} P$. Then the faces of $\hat{P}$ are exactly the following:*

- *faces of P that do not contain v;*
- *faces of the form $\operatorname{conv}\{v, G\}$, where*
  - *G is a face of P not containing v, and*
  - *G is contained in a facet of P that contains v;*

Furthermore, for every face F of P containing v, there is a facet F' of F that yields a face $\operatorname{conv}\{v, F'\}$ of the second type.

Proof. All claims except the last statement are from [14, 5.2.2]. The last statement follows by considering $\operatorname{conv}(\operatorname{vert} F \setminus \{v\})$, which does not contain v, and is either a facet of F or contains a facet of F. □

Using the inclusion-reversing correspondence between faces of a polytope and faces of its dual, the following lemma shows what pushing a facet will do:

Lemma 9 *Suppose $\hat{P} \subset \Re^d$ is a d-polytope obtained from the d-polytope P by pushing a facet $F = h \cap P$ of P. Then the faces of $\hat{P}$ correspond to the following:*

- *faces of P that do not meet F;*
- *faces of the form $\hat{h} \cap G$, where*
  - *G is a face not contained in F, and*
  - *G contains a vertex of F.*

Furthermore, for every face H of F, there is a face G of P such that H is a facet of G, and G yields a face $\hat{h} \cap G$ of the second type.

Proof. The lemma follows directly from the previous one, and duality. □

Lemma 10 *For a line-free polyhedral set P of dimension d and with n facets, the triangulation $\Delta(P)$ has $O(n^{\lfloor d/2 \rfloor})$ simplices, as $n \to \infty$.*

Proof. If P is a line-free polyhedral set with the origin in its interior, the dual P^* is a polytope, and the operation of pushing the ideal facet of P can be defined using the dual operation of pulling the origin, which will be the vertex of P^* corresponding to $\operatorname{id} P$. The analogue of Lemma 9 holds for the case of pushing $\operatorname{id} P$, using duality.

From Lemma 7 and the above discussion, it suffices to show that if $\hat{P}$ is the result of pushing facet $F = h \cap P$ of P, then $|\Delta(\hat{P})| \geq |\Delta(P)|$. Lemma 9 implies that for every face of P, there is a face of $\hat{P}$; that is, there is an injective mapping $m : \mathcal{B}(P) \mapsto \mathcal{B}(\hat{P})$. The lemma follows using induction on dimension. □

Lemma 11 *The complex $\Delta(\mathcal{V}(R))$ has $O(r^{\lceil d/2 \rceil})$ regions, and $O(r^{\lceil d/2 \rceil} \log r)$ time suffices for its construction. The constant factors are $e^{O(d \log d)}$.*

Proof. Without loss of generality, we need consider only a bound on the size of $\Delta(\mathcal{V}(R))$ when $d = \dim R$.

Let Z_R denote the polytope $\operatorname{conv} \Upsilon(R)$, dual to $\mathcal{V}(R)$ from the discussion of §2. It is easy to see that $\hat{Z_R}$ has the same facial lattice structure as $\mathcal{V}(R)$ (is combinatorially equivalent). It follows that $\Delta(\hat{Z_R})$ has the same number of simplices as $\Delta(\mathcal{V}(R))$. Lemma 10 then gives the desired bound.

The time required to construct $\Delta(\mathcal{V}(R))$ is dominated by the time necessary for determining $\mathcal{V}(R)$: the projection P_R can be computed in $O(r)$ time, and $\mathcal{V}(R)$ can be triangulated in time linear in the number of its faces.

Several algorithms are known for computing $\mathcal{V}(R)$ in $O(r \log r)$ time when $2 = \dim R$ [12], [22]. As noted in §2, the computation of $\mathcal{V}(R)$ can be reduced to the computation of the convex hull of $\Upsilon(R)$. This can be done in $O(r^2 + r^{\lceil d/2 \rceil} \log r)$ time [25]. For $d > 2$, this is $O(r^{\lceil d/2 \rceil} \log r)$. □

4 Candidate regions and sets for Voronoi diagram triangulations

4.1 Candidate regions have a simple description The following theorem characterizes the candidate regions of line-free simplices in $\Delta(\mathcal{V}(R))$. The general case is considered in Theorem 13 below.

Theorem 12 *Let line-free $A \in \Delta(\mathcal{V}(R))$, with $A \subset \mathcal{V}_q$ for a site q. Then the candidate region $C(A)$ is*

$$C(A) = \bigcup_{a \in \operatorname{vert} A} B_{a,q}.$$

(As noted above, the points in $\operatorname{vert} A$ are vertices of Voronoi regions, and the balls are either Delaunay balls or halfspaces corresponding to convex hull facets. The ideal vertices of A are ideal vertices of $\mathcal{V}_q$, which correspond to unbounded edges of $\mathcal{V}_q$ that are normal to a convex hull facet containing q.)

Proof. It suffices to show that $C(A) \subseteq \cup_{a \in \operatorname{vert} A} B_{a,q}$, as the reverse inclusion follows by definition. That is, we must show that if $x \in A$, then

$$C(x) = B_{x,q} \subset \bigcup_{a \in \operatorname{vert} A} B_{a,q}.$$

Suppose $y \notin B_{a,q}$ for all $a \in \operatorname{vert} A$. The theorem follows if this condition implies that $y \notin B_{x,q}$.

From §2, $y \notin B_{a,q}$ if and only if $a_h \cdot (\Upsilon(y) - \Upsilon(q)) \geq 0$. By Lemma 2, $A = \operatorname{conv} \operatorname{vert} A$, and so

$$x_h = \sum_{a \in \operatorname{vert} A} \alpha_a a_h$$

for some $\alpha_a \geq 0$, not all zero. Therefore,

$$x_h \cdot (\Upsilon(y) - \Upsilon(q)) = \left(\sum_{a \in \operatorname{vert} A} \alpha_a a_h \right) \cdot (\Upsilon(y) - \Upsilon(q)) = \sum_{a \in \operatorname{vert} A} \alpha_a a_h \cdot (\Upsilon(y) - \Upsilon(q)) \geq 0.$$

This implies $y \notin B_{x,q}$, and the lemma follows. □

Note that the only property of the line-free polyhedral set A on which the proof depends is that $A \subset \mathcal{V}_q$, so that an analogous result holds for any such polyhedral set.

This characterization of $C(A)$ must be extended to the case where A is not necessarily line-free.

Theorem 13 *Let $A \in \Delta(\mathcal{V}(R))$, with $A \subset \mathcal{V}_q$ for a site q. Then the candidate region $C(A)$ is*

$$C(A) = C(A \cap \operatorname{aff} R) \cup (F^d \setminus \operatorname{aff} R).$$

Proof. From the discussion following Lemma 6, A has the form $(A \cap \operatorname{aff} R) + (\operatorname{aff} R)^\perp$. Also from that discussion, $\operatorname{aff} R \subseteq S_{c,q}$ for ideal c if and only if $c \in (\operatorname{aff} R)^\perp$. Since $C(A) \supseteq B_{c,q}$ for all $c \in \operatorname{id}(\operatorname{aff} R)^\perp$, we have

$$C(A) \supseteq F^d \setminus \bigcap_{c \in \operatorname{id}(\operatorname{aff} R)^\perp} S_{c,q},$$

that is, $C(A) \supseteq F^d \setminus \operatorname{aff} R$.

The theorem now follows by showing that $C(A) \cap \operatorname{aff} R \subseteq C(A \cap \operatorname{aff} R) \cap \operatorname{aff} R$. Let $z \in A$. First, suppose z is a real point. Then by the discussion following Lemma 6, $z = x + y$, for $x \in \operatorname{re} A \cap \operatorname{aff} R$ and $y \in \operatorname{re}(\operatorname{aff} R)^\perp$. Since $[x_r/x_s + y_r/y_s; 1]$ is a homogeneous representation for z, we have, for $w \in \operatorname{aff} R$,

$$z_h \cdot (\Upsilon(w) - \Upsilon(q)) = [x_r/x_s; 1] \cdot (\Upsilon(w) - \Upsilon(q)) + [y_r/y_s; 0] \cdot (\Upsilon(w) - \Upsilon(q)),$$

but $[y_r/y_s; 0] \in \mathrm{id}(\operatorname{aff} R)^\perp$, and so

$$z_h \cdot (\Upsilon(w) - \Upsilon(q)) < 0 \text{ if and only if } x_h \cdot (\Upsilon(w) - \Upsilon(q)) < 0.$$

That is, $C(z) \cap \operatorname{aff} R = C(x) \cap \operatorname{aff} R$. If z is an ideal point not in $(\operatorname{aff} R)^\perp$, then $z_h = \alpha x_h + \beta y_h$, for some $x \in \mathrm{id}(A \cap \operatorname{aff} R)$, $y \in \mathrm{id}(\operatorname{aff} R)^\perp$, $\alpha > 0$, and $\beta \geq 0$. In this case, similarly, $C(z) \cap \operatorname{aff} R = C(x) \cap \operatorname{aff} R$. Finally, if $z \in \mathrm{id}(\operatorname{aff} R)^\perp$, then $C(z) = B_{z,q}$, and $S_{z,q} = \operatorname{aff} R$, so that $C(z) \cap \operatorname{aff} R = \emptyset$. Thus $C(A)$ and $C(A \cap \operatorname{aff} R)$ agree on $\operatorname{aff} R$, and the theorem follows. □

4.2 Reporting all closest sites

While an RPO tree allows a closest site to a query point to be found, sometimes it is of interest to find all the sites closest to a given point. In this case, the distinction between $C(A)$ and $\overline{C}(A)$ becomes important. For example, suppose the sites are all on the surface of a sphere $S_{c,q}$, and the query point is the center of the sphere c. Here the set $C(c) \cap S$ will be empty, but the set $\overline{C}(c) \cap S = S$. To handle such situations, it will be shown below that, roughly speaking, for most points $x \in A \in \Delta(\mathcal{V}(R))$, $\overline{C}(x)$ is contained in $C(A)$.

Theorem 14 *Under the conditions of Theorem* 12*, let* $x \in A$*, and* F *be the face of* A *with* $x \in \operatorname{relint} F$*. Let* $F' = F \cap \operatorname{aff} R$*. Then* $\overline{C}(x) \subset C(A) \bigcup C_\cap(F)$*, where*

$$C_\cap(F) = \bigcap_{a \in \operatorname{vert} F'} S_{a,q} \cap \operatorname{aff} R.$$

(The existence and uniqueness of such a face F is readily established using elementary properties of polytopes, as given in [14, §2.6]. Note that when x is in the interior of some $\mathcal{V}_q$, the associated region $C_\cap(\mathcal{V}_q)$ is trivial: it is easy to show that $C_\cap(A) = \{q\}$. Note also that in the two-dimensional case, the region $C_\cap(e)$ for some Voronoi edge e is simply the intersection of the two Delaunay circles of the endpoints of e. This intersection contains only the two sites of R that define e.)

Proof. It is easy to show that $x \in \operatorname{relint} F$ implies that $x = z + \sum_{a \in \operatorname{vert} F'} \alpha_a a$, for some $z \in (\operatorname{aff} R)^\perp$ and some α_a all strictly greater than zero. (This holds necessarily only if $F \cap \operatorname{aff} R$ is a simplex.) For a point y, reasoning similar to that in the proof of Theorem 12 implies that when $y \notin C(A)$, we have $a_h \cdot (\Upsilon(y) - \Upsilon(q)) \geq 0$ for all $a \in \operatorname{vert} F'$. Thus,

$$x_h \cdot (\Upsilon(y) - \Upsilon(q)) = \sum_{a \in \operatorname{vert} F'} \alpha_a a_h \cdot (\Upsilon(y) - \Upsilon(q)) \geq 0.$$

If $x_h \cdot (\Upsilon(y) - \Upsilon(q)) > 0$ then $y \notin \overline{C}(x)$, so suppose $x_h \cdot (\Upsilon(y) - \Upsilon(q)) = 0$. Since $\alpha_a > 0$ and $a_h \cdot (\Upsilon(y) - \Upsilon(q)) \geq 0$ for all $a \in \operatorname{vert} F'$, we must have $a_h \cdot (\Upsilon(y) - \Upsilon(q)) = 0$ for all $a \in \operatorname{vert} F'$. The y for which this holds are precisely those in $C_\cap(F)$. □

When all sites closest to a query point are desired, the function *Make_RPO_Tree* is modified so that for each face F of a region $A \in \Delta(\mathcal{V}(R))$, the sites $F.sites = C_\cap(F) \cap S$ are stored for the node v with $v.node = A$. When answering a query, the variable *current_closest* represents a set of sites, the sites so far found closest to the query point p. At each step of *Answer_Query*, the face F of A with query point $p \in \operatorname{relint} F$ is found, and the distance of the sites in $F.sites$ to p is compared with the distance of those in *current_closest*. (Note that all sites in $F.sites$ are equidistant from p, and similarly for *current_closest*.) If the sites in $F.sites$ are the same distance as those in *current_closest*, they are added to the set *current_closest*. If they are closer, they replace that set, and if they are farther, that set is unchanged. If *current_closest* is maintained as a list of lists of sites, this updating operation requires constant time.

4.3 Candidate sets are likely to be small

The theorem below implies that, with probability $1/2$, the candidate sets generated by *Make_RPO_Tree* all contain few sites. This ensures that an RPO tree can be created that has height $O(\log n)$, and allows a bound on the tree's total size.

As a warm-up, here is a lemma regarding the Delaunay balls.

Lemma 15 *For $S \subset E^d$, let $R \subset S$ be a random sample (without replacement) of size r. Let P_α be the probability that any open Delaunay ball B has $|S \cap B| > \alpha n$. Then $P_\alpha \leq 1/2$, for*

$$\alpha \geq \frac{\ln\left(2\binom{r}{d+1}\right)}{r-d-1}.$$

That is, $1 - P_\alpha \geq 1/2$, where $1 - P_\alpha$ is the probability that for all Delaunay balls B, it holds that $|S \cap B| \leq \alpha n$.

Proof. Suppose that $R' \subset R$ is the set of the first $d+1$ samples taken. If $d = \dim R'$, then the sphere containing R' defines an open ball B. Now suppose $|S \cap B| > \alpha n$. Then the probability that none of the remaining $r-d-1$ samples are taken from $S \cap B$ is bounded above by $(1-\alpha)^{r-d-1}$. That is, with probability at least $1-(1-\alpha)^{r-d-1}$, B will not be a Delaunay ball of $\mathcal{V}(R)$. For sufficiently large α, the latter probability is large.

Now let X be the set of all subsets of R of size $d+1$, let $B_{R'}$ be the open ball defined by subset $R' \in X$, and let $B(X)$ be the set of open balls defined by these subsets. Let $B_\alpha(X) \subset B(X)$ be the set of all such balls B satisfying $|S \cap B| > \alpha n$. If no ball $B \in B_\alpha(X)$ satisfies $R \cap B = \emptyset$, then every ball in $B(X)$ that does not contain any sample sites must not be in $B_\alpha(X)$. That is, every Delaunay ball of R must contain a proportion of sites smaller than α.

What is an upper bound on the probability P_α that at least one $B \in B(X)$ has $B \in B_\alpha(X)$ and $R \cap B = \emptyset$? For a given ball $B \in B(X)$, the joint probability of these two conditions is no more than the conditional probability that $R \cap B = \emptyset$ given $B \in B_\alpha(X)$. The latter probability is the same as that for the ball defined by the first $d+1$ sample sites. Since the probability of the union of a set of events is not more than the sum of the probabilities of the individual events, we have

$$P_\alpha = \text{Prob}\{\exists R' \in X \mid B_{R'} \in B_\alpha(X) \text{ and } R \cap B_{R'} = \emptyset\} < \binom{r}{d+1}(1-\alpha)^{r-d-1}.$$

When $\alpha \geq \ln(2\binom{r}{d+1})/(r-d-1)$, this probability is no more than 1/2, using the relation $-\ln(1-\alpha) \geq \alpha$ for $0 \leq \alpha < 1$. □

This lemma is not a proof of the desired result for general $C(A)$, since not all regions $C(A)$ are the union of Delaunay balls. However, the proof of the following theorem is quite similar to that of the lemma.

Theorem 16 *For $S \subset E^d$, let $R \subset S$ be a random sample of size r. Let P_α be the probability that any one of the regions $A \in \Delta(\mathcal{V}(R))$ has $|S \cap C(A)| > \alpha n$. Then $P_\alpha \leq 1/2$, for*

$$\alpha \geq \alpha_{r,d} = \frac{(d+1)\ln\left((d+1)^2\binom{r}{d+1}\right)}{r-d-1}.$$

That is, $1 - P_\alpha \geq 1/2$, where $1 - P_\alpha$ is the probability that for all $A \in \Delta(\mathcal{V}(R))$, it holds that $|S \cap C(A)| \leq \alpha n$.

Proof. By Theorems 12 and 13, for $A \in \Delta(\mathcal{V}(R))$ with $A \subseteq \mathcal{V}_q$,

$$C(A) = \bigcup_{a \in \text{vert}(A \cap \text{aff } R)} [B_{a,q} \cup (F^d \setminus \text{aff } R)].$$

The simplex $A \cap \text{aff } R$ has at most $1 + \dim R$ vertices, so the number of regions making up this union is no more than $d+1$. Let $\alpha' = \alpha/(d+1)$. The condition $|S \cap C(A)| > \alpha n$ thus implies that $|S \cap C(I)| > \alpha' n$, where I is a region $B_{a,q} \cup (F^d \setminus \text{aff } R)$. It suffices to prove that with probability 1/2, all such regions contain no more than $\alpha' n$ sites.

A region I may have a real or a ideal. If a is real, there is a set of $1 + \dim R$ sites $R' \subset R$ such that I is the union of $F^d \setminus \text{aff } R'$ with the relatively open ball defined by the $(\dim R)$-sphere in aff R' that contains R'. If a is ideal, there is a set of $\dim R$ sites R' and a site $s \in R$ such that I is the union of $F^d \setminus \text{aff}(R' \cup \{s\})$ with the open half-flat of $\text{aff}(R' \cup \{s\})$ that is bounded by aff R' and that does not contain s.

Let X be the set of all nonempty subsets of R of size $d+1$ or less, together with the set of pairs (R', s) where $R' \subset R$, $s \in R$, and $1 \leq |R'| \leq d$. Let I_x be the region corresponding to $x \in X$ as above, and let $I(X)$ be the set of regions corresponding to the elements of X. Let $I_{\alpha'}(X)$ be the subset of $I(X)$

containing regions $I \in I(X)$ satisfying $|S \cap I| > \alpha' n$. We have, for any given region $I \in I(X)$,

$$\text{Prob}\{R \cap I = \emptyset \text{ given } I \in I_{\alpha'}(X)\} \le (1-\alpha')^{r-d-1}.$$

The probability

$$\text{Prob}\{\exists x \in X \mid I_x \in I_{\alpha'}(X) \text{ and } R \cap I_x = \emptyset\}$$

is greater than P_α, and is bounded above by $(1-\alpha')^{r-d-1}$ times the size of X. It is easy to see that $|X| = (d+1)\binom{r}{d+1} + d(d+1)\binom{r}{d+1}$, or $|X| = (d+1)^2\binom{r}{d+1}$. The theorem follows, using manipulations as in the lemma above. □

5 Time bounds for Make_RPO_Tree and Answer_Query

To bound the time needed for *Make_RPO_Tree*, we will first consider the work done by the procedure, aside from the recursive calls, and then bound the work for those calls.

Lemma 17 *Let $t(n)$ denote the expected time required by* Make_RPO_Tree. *Then $t(n)$ satisfies*

$$t(n) \le K_1 n r^{\lceil d/2 \rceil} \log r + K_2 r^{\lceil d/2 \rceil} t\left(K_3 n \frac{\ln\ r}{r}\right),$$

when $n > K$. The constants K_1 and K_2 are at most exponential in $O(d \log d)$, and K_3 is $(d+1)^2 + O(1/\log r)$, as $r \to \infty$.

Proof. From Theorem 16, the **repeat**–**until** loop for determining a suitable $\Delta(\mathcal{V}(R))$ will end after two iterations on the average, and require $O(r^{\lceil d/2 \rceil} \log r)$ each iteration, by Lemma 11.

The other operations in *Make_RPO_Tree* require $O(n)$ or $O(r)$ time, except for the recursive calls and the determination, for each $A \in \Delta(\mathcal{V}(R))$, of $S \cap C(A)$.

From the proof of Lemma 7 and precise bounds on the number of vertices of a d-polytope with r facets [14, §4.7], the constants K_1 and K_2 are dominated by $d!$, which is exponential in $O(d \log d)$.

From Theorem 16, the size of each subproblem is $\alpha_{r,d} n$. The bound for K_3 follows from the value of $\alpha_{r,d}$ and elementary approximations. □

Theorem 18 *The expected time $t(n)$ required by* Make_RPO_Tree *is bounded by $t(n) = O(n^{\lceil d/2 \rceil (1+\epsilon)})$, as $n \to \infty$, where*

$$\epsilon = \frac{\ln(K_3 \ln r) + (\ln K_2)/\lceil d/2 \rceil}{\ln(r/K_3 \ln r)},$$

for fixed r and d.

Proof. By "unrolling" the recurrence for $t(n)$ to depth $m = \ln(n/K)/\ln(r/K_3 \ln r)$, we have

$$t(n) = O(nr^{\lceil d/2 \rceil} \log r(K_2 K_3 r^{\lceil d/2 \rceil - 1} \ln\ r)^m),$$

and the desired expression follows by algebraic manipulations. □

Theorem 19 *The worst-case time required by procedure* Answer_Query *is bounded by*

$$K + K_2 r^{\lceil d/2 \rceil} \ln(n/K)/\ln(r/K_3 \ln r).$$

This is $O(\log n)$ *as* $n \to \infty$*, for fixed* r *and* d*.*

Proof. This is just the work in searching through the children of an RPO tree, times the depth of such a tree. □

6 Conclusions

We have seen that a simple, natural approach to the post-office problem may be used to gain great improvements in asymptotic efficiency over methods previously known for $d > 3$. In addition, this approach has an advantage of conceptual and programming simplicity over previous asymptotically fast methods for $d \leq 3$.

The approach given here may be used to yield fast algorithms for other proximity problems. For example, suppose a convex three-dimensional polytope P is given, and a data structure is to be found such that given a plane h with $h \cap P = \emptyset$, the vertex of P closest to h is to be determined quickly. This problem is equivalent to determining the point of $\operatorname{vert} P$ closest to the ideal point normal to h, and is also equivalent to linear programming in 3-D with multiple objective functions. The problem may be solved with nearly linear preprocessing and logarithmic query time using an approach analogous to that given in this paper.

After the preliminary report of these results [5], later work has shown that these ideas have applications in many other areas of discrete and computational geometry, such as arrangement searching, determining the separation of polytopes, constructing order k Voronoi diagrams [6], computing line-segment intersections, bounding ($\leq k$)-sets in E^d [7], computing the diameter of a point set in E^3, incremental construction of geometric structures [8], and triangulating simple polygons [9]. Independently of this work, the concept of the Vapnik-Chervonenkis (VC) dimension [28] has been applied to, for example, the problem of halfspace range queries, resulting in a randomized algorithm for the construction of a data structure for such queries [16]. This concept has also been applied to questions of learnability [2]. While apparently not equivalent, the two approaches (the VC dimension and that of this paper) are similar in spirit, and provide a useful means of applying divide-and-conquer to computational geometry.

Appendix A

Proofs of three technical lemmas are given below.

LEMMA 1. *Let C be a polyhedral pointed cone with apex a. Let h be a supporting hyperplane of C with $h \cap C = \{a\}$. Let ν be a normal vector to h contained in the same halfspace containing C. Then $P = C \cap (h + \nu)$ is a polytope.*

Proof. (Note that such a hyperplane h exists because a is a face.) Since C and $h + \nu$ are polyhedral sets, it follows that their intersection is a polyhedral set. It remains to show that P is bounded. If not, then P contains a ray, by [14, 2.5.1]. Such a ray has the form $\mathrm{ray}_z\, y$, where $z, y \in P$ and $\nu \cdot (y - z) = 0$. Since C is a cone, the point $(x - a)/\|x - a\| + a \in \mathrm{ray}_a\, x$ is in C, for every $x \in \mathrm{ray}_z\, y$. As $\|x\| \to \infty$, with x on $\mathrm{ray}_z\, y$, the points $(x - a)/\|x - a\| + a$ converge to $(y - z)/\|y - z\| + a$. Since C is closed, this point is in C. But $(y - z)/\|y - z\| + a \in h$, contradicting the choice of h. □

LEMMA 2. *If $P \subset E^d$ is a line-free d-polyhedral set then $P = \mathrm{conv}\,\mathrm{vert}\,P$.*

Proof. The lemma is true for polytopes by [14, 2.4.5]. The unbounded case will first be considered for polyhedral cones, and then in general.

If C is a line-free polyhedral cone, then as mentioned above, C is pointed, so that Lemma 2 applies to C. Using the correspondence above between $x \in P$ and $x' \in \mathrm{id}\,C$, the fact that $P = \mathrm{conv}\,\mathrm{vert}\,P$ implies that $\mathrm{id}\,C = \mathrm{conv}\,\mathrm{vert}\,\mathrm{id}\,C$. Since

$$C = \bigcup_{x \in \mathrm{id}\,C} \mathrm{conv}\{a, x\},$$

it follows that for any $y \in C$,

$$y_h = \alpha_a a_h + \sum_{x \in \mathrm{vert}\,\mathrm{id}\,C} \alpha_x x_h,$$

for some $\alpha_a, \alpha_x \geq 0$, $x \in \mathrm{vert}\,\mathrm{id}\,C$. That is, $C \subset \mathrm{conv}\,\mathrm{vert}\,C$. It is easy to show that $C \supset \mathrm{conv}\,\mathrm{vert}\,C$, so the lemma follows for line-free polyhedral cones.

To prove the lemma for general line-free polyhedral sets, we appeal to [14, 2.5.6], which directly implies that a line-free polyhedral set P can be expressed as $P = \mathrm{cc}\,P + \mathrm{conv}\,\mathrm{re}\,\mathrm{vert}\,P$. Since $\mathrm{cc}\,P$ is line-free if P is, the relation $\mathrm{cc}\,P = \mathrm{conv}\,\mathrm{vert}\,\mathrm{cc}\,P$ holds. The lemma follows. □

LEMMA 4. *For a line-free polyhedral region P, if $v \in \mathrm{vert}\,P$ and $a \in P$, then the point $b = \mathrm{ray}_v\, a \cap \mathrm{relbd}\,P$ is on a facet of P that does not contain v.*

Proof. (The relative boundary of P is generalized to include $\mathrm{id}\,P$.) Since $\mathrm{relbd}\,P$ is the union of the facets of P [14, 2.6.3], b is on some facet of P. Suppose $v \notin \mathrm{id}\,P$. Then the lemma follows by induction on dimension: suppose v and b are on the same facet F. Then assuming the lemma for the polytope F, b is on some facet of F not containing v. Such a facet of F is the intersection of F with another facet F' of P [14, 2.6.4], and so $b \in F'$ but $v \notin F'$. Suppose $v \in \mathrm{id}\,P$. If a is a real point, then so is b, and the lemma follows. If $a \in \mathrm{id}\,P$, then we map v, a, and so b to a polytope as in Lemma 1, and the lemma follows by the above argument. □

Acknowledgments

I thank Jon Bentley, Brian Kernighan, Danny Sleator, and Chris Van Wyk for their comments on an earlier draft of this paper. I also thank the referees for their helpful comments on another draft. Finally, I thank Chris Van Wyk for comments on yet another draft.

References

[1] J. L. BENTLEY, B. WEIDE, AND A. C. YAO, *Optimal expected-time algorithms for closest-point problems*, ACM Trans. Math. Software, 6 (1982), pp. 563–579.

[2] A. BLUMER, A. EHRENFEUCHT, D. HAUSSLER, AND M. WARMUTH, *Classifying learnable geometric concepts with the Vapnik-Chervonenkis dimension*, in Proc. 17th Annual ACM Symposium on Theory of Computing, 1986.

[3] K. Q. BROWN, *Voronoi diagrams from convex hull*, Inform. Proc. Lett., 9 (1979), pp. 223–228.

[4] B. CHAZELLE, *How to search in history*, Inform. and Control, 64 (1985), pp. 77–99.

[5] K. L. CLARKSON, *A probabilistic algorithm for the post office problem*, in Proc. 17th Annual ACM Symposium on Theory of Computing, 1985, pp. 75–184.

[6] ——, *New applications of random sampling in computational geometry*, Discrete Comput. Geometry, 2 (1987), pp. 195–222.

[7] ——, *Random sampling in computational geometry,* II, in Proc. Fourth Symposium on Computational Geometry, 1988, pp. 1–11.

[8] K. L. CLARKSON AND P. W. SHOR, *Algorithms for diametral pairs and convex hulls that are optimal, randomized, and incremental*, in Proc. Fourth Symposium on Computational Geometry, 1988, pp. 12–17.

[9] K. L. CLARKSON, R. E. TARJAN, AND C. J. VAN WYK, *A fast Las Vegas algorithm for triangulating a simple polygon*, in Proc. Fourth Symposium on Computational Geometry, 1988, pp. 18–22.

[10] D. Dobkin and R. J. Lipton, *Multidimensional searching problems*, SIAM J. Comput., 5 (1976), pp. 181–186.

[11] H. Edelsbrunner, L. Guibas, and J. Stolfi, *Optimal point location in a planar subdivision*, SIAM J. Comput., 15 (1986), pp. 317–340.

[12] S. J. Fortune, *A sweepline algorithm for Voronoi diagrams*, in Proc. Second Symposium on Computational Geometry, 1986, pp. 313–322.

[13] E. N. Gilbert, *Random subdivisions of space into crystals*, Ann. Math. Statist., 33 (1962), pp. 958–972.

[14] B. Grünbaum, *Convex Polytopes*, John Wiley, New York, 1967.

[15] L. J. Guibas, L. Ramshaw, and J. Stolfi, *A kinetic framework for computational geometry*, in Proc. 24th IEEE Symposium on Foundations of Computer Science, 1983, pp. 100–111.

[16] D. Haussler and E. Welzl, *Epsilon-nets and simplex range queries*, Discrete Comput. Geometry, 2 (1987), pp. 127–151.

[17] D. Kirkpatrick, *Optimal search in planar subdivisions*, SIAM J. Comput., 12 (1983), pp. 28–35.

[18] V. Klee, *On the complexity of d-dimensional Voronoi diagrams*, Arch. Math., 34 (1980), pp. 75–80.

[19] R. J. Lipton and R. E. Tarjan, *Applications of a planar separator theorem*, in Proc. 18th IEEE Symposium on Foundations of Computer Science, 1977, pp. 162–169.

[20] P. McMullen, *The maximum number of faces of a convex polytope*, Mathematika, 17 (1970), pp. 179–184.

[21] D. E. Muller and F. P. Preparata, *Finding the intersection of two convex polyhedra*, Theoret. Comput. Sci., 7 (1978), pp. 217–236.

[22] F. P. Preparata and M. I. Shamos, *Computational Geometry: An Introduction*, Springer-Verlag, New York, Berlin, 1985.

[23] E. G. Rees, *Notes on Geometry*, Springer-Verlag, New York, Berlin, 1983.

[24] R. Seidel, *On the size of the closest point Voronoi diagrams*, Tech. Rep. 94, Institut für Informationsverarbeitung, Technische Universität Graz und Osterreichissche Computergesellschaft, Austria, 1982.

[25] ——, *Constructing higher-dimensional convex hulls at logarithmic cost per face*, in Proc. 18th Annual ACM Symposium on Theory of Computing, 1986, pp. 404–413.

[26] M. I. Shamos, *Computational Geometry*, Ph.D. thesis, Yale University, New Haven, CT, 1978.

[27] J. Stolfi, *Oriented projective geometry*, in Proc. Third Symposium on Computational Geometry, 1987, pp. 76–85.

[28] V. N. Vapnik and A. Y. A. Chervonenkis, *On the uniform convergence of relative frequencies of events to their probabilities*, Theory Probab. Appl., 16 (1971), pp. 264–280.