1

Cuttings

Bernard Chazelle
Princeton University

1.1 Introduction

For divide-and-conquer purposes, it is often desirable to organize a set S of n numbers into a sorted list, or perhaps to partition it into two equal-sized groups with no element in one group exceeding any element in the other one. More generally, we might wish to break up S into k groups of size roughly n/k, with again a total ordering among the distinct groups. In the first case we sort; in the second one we compute the median; in the third one we compute quantiles. This is all well known and classical. Is it possible to generalize these ideas to higher dimension? Surprisingly the answer is yes. A geometric construction, known as an *ε-cutting*, provides a space partitioning technique that extends the classical notion of selection to any finite dimension. It is a powerful, versatile data structure with countless applications in computational geometry.

Let H be a set n hyperplanes in $\mathbf{R}^d$. Our goal is to divide up $\mathbf{R}^d$ into simplices, none of which is cut by too many of the n hyperplanes. By necessity, of course, some of the simplices need to be unbounded. We choose a parameter $\varepsilon > 0$ to specify the coarseness of the subdivision. A set $\mathcal{C}$ of closed full-dimensional simplices is called an ε-cutting for H (Fig. 1.1) if:

(i) the union of the simplices is $\mathbf{R}^d$, and their interiors are mutually disjoint;
(ii) the interior of any simplex is intersected by at most εn hyperplanes of H.

Historically, the idea of using sparsely intersected simplices for divide and conquer goes back to Clarkson [10] and Haussler and Welzl [15], among others. The definition of an ε-cutting given above is essentially due to Matoušek [18]. Efficient but suboptimal constructions were given by Agarwal [1, 2] for the two-dimensional case and Matoušek [17, 18, 21] for arbitrary dimension. The optimal ε-cutting construction cited in the theorem below, due to Chazelle [4], is a simplification of an earlier design by Chazelle and Friedman [7].

THEOREM 1.1 *Given a set H of n hyperplanes in $\mathbf{R}^d$, for any $0 < \varepsilon < 1$, there exists an ε-cutting for H of size $O(\varepsilon^{-d})$, which is optimal. The cutting, together with the list*

0-8493-8597-0/01/$0.00+$1.50

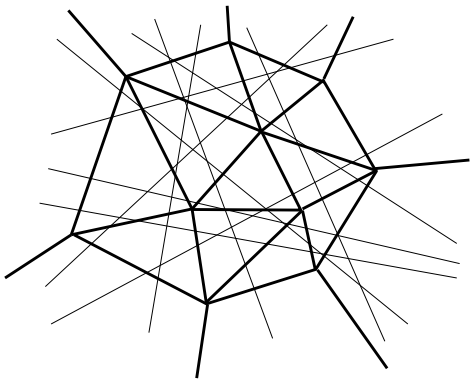

FIGURE 1.1: *A two-dimensional cutting.*

of hyperplanes intersecting the interior of each simplex, can be found deterministically in $O(n\varepsilon^{1-d})$ *time.*

1.2 The Cutting Construction

This section explains the main ideas behind the proof of Theorem 1.1. We begin with a quick overview of geometric sampling theory. For a comprehensive treatment of the subject, see [6, 23].

1.2.1 Geometric Sampling

A *set system* is a pair $\Sigma = (X, \mathcal{R})$, where X is a set and $\mathcal{R}$ is a collection of subsets of X. In our applications, $X \subset \mathbf{R}^d$ and each $R \in \mathcal{R}$ is of the form $X \cap f(K)$, where K is a fixed region of $\mathbf{R}^d$ and f is any member of a fixed group F of transformations. For example, we might consider n points in the plane, together with the subsets lying inside any triangle congruent to a fixed triangle.

Given $Y \subseteq X$, we define the set system "induced by Y" to be $(Y, \mathcal{R}|_Y)$, with $\mathcal{R}|_Y = \{\, Y \cap R \,|\, R \in \mathcal{R} \,\}$. The *VC-dimension* (named for Vapnik and Chervonenkis [28]) of Σ is defined as the maximum size of any Y such that $\mathcal{R}|_Y = 2^Y$. For example, the VC-dimension of the infinite geometric set system formed by points in the plane and halfplanes is 3. The *shatter function* $\pi_{\mathcal{R}}(m)$ of the set system $\Sigma = (X, \mathcal{R})$ is the maximum number of subsets in the set system $(Y, \mathcal{R}|_Y)$ induced by any $Y \subseteq X$ of size m. If $\pi_{\mathcal{R}}(m)$ is bounded by cm^d, for some constants $c, d > 0$, then the set system is said to have a *shatter function exponent* of at most d. It was shown in [26, 27, 28] that, if the shatter function exponent is $O(1)$, then so is the VC-dimension. Conversely, if the VC-dimension is $d \geq 1$ then, for any $m \geq d$, $\pi_{\mathcal{R}(m)} < (em/d)^d$.

We now introduce two fundamental notions: ε-nets and ε-approximations. For any $0 < \varepsilon < 1$, a set $N \subseteq X$ is called an *ε-net* for a finite set system $(X, \mathcal{R})$ if $N \cap R \neq \emptyset$ for any $R \in \mathcal{R}$ with $|R|/|X| > \varepsilon$. A finer (but more costly) sampling mechanism is provided by an

ε-approximation for $(X, \mathcal{R})$, which is a set $A \subseteq X$ such that, given any $R \in \mathcal{R}$,

$$\left| \frac{|R|}{|X|} - \frac{|A \cap R|}{|A|} \right| \leq \varepsilon.$$

Some simple structural facts about nets and approximations:

LEMMA 1.1 If X_1, X_2 are disjoint subsets of X of the same size, and A_1, A_2 are same-size ε-approximations for the subsystems induced by X_1, X_2 (respectively), then $A_1 \cup A_2$ is an ε-approximation for the subsystem induced by $X_1 \cup X_2$.

LEMMA 1.2 If A is an ε-approximation for $(X, \mathcal{R})$, then any ε'-approximation (resp. -net) for $(A, \mathcal{R}|_A)$ is also an $(\varepsilon + \varepsilon')$-approximation (resp. -net) for $(X, \mathcal{R})$.

In the absence of any restrictive assumption on the set system, it is natural to expect the sample size to depend on both the desired accuracy and the size of the set system itself.

THEOREM 1.2 *Given a set system $(X, \mathcal{R})$, where $|X| = n$ and $|\mathcal{R}| = m$, for any $1/n \leq \varepsilon < 1$, it is possible to find, in time $O(nm)$, an ε-net for $(X, \mathcal{R})$ of size $O(\varepsilon^{-1} \log m)$ and an ε-approximation for $(X, \mathcal{R})$ of size $O(\varepsilon^{-2} \log m)$.*

If we assume bounded VC-dimension, everything changes. In fact the key result in geometric sampling theory is that, for any given level of accuracy, the sample size need not depend on the size of the set system.

In practice, geometric set systems often are "accessible" via an *oracle* function that takes any $Y \subseteq X$ as input and returns the list of sets in $\mathcal{R}|_Y$ (each set represented explicitly). We assume that the time to do that is $O(|Y|^{d+1})$, which is linear in the maximum possible size of the oracle's output, where d is the shatter function exponent. For example, in the case of points and disks in the plane, we have $d = 3$, and so this assumes that, given n points, we can enumerate all subsets enclosed by a disk in time $O(n^4)$. To do this, enumerate all k-tuples of points ($k \leq 3$) and, for each tuple, find which points lie inside the smallest disk enclosing the k points. The main result below is stated in terms of the shatter function exponent d, but the same results hold if d denotes the VC-dimension.

THEOREM 1.3 *Given a set system $(X, \mathcal{R})$ of shatter function exponent d, for any $\varepsilon \leq 1/2$, an ε-approximation for $(X, \mathcal{R})$ of size $O(d\varepsilon^{-2} \log d\varepsilon^{-1})$ and an ε-net for $(X, \mathcal{R})$ of size $O(d\varepsilon^{-1} \log d\varepsilon^{-1})$ can be computed in time $O(d)^{3d}(\varepsilon^{-2} \log d\varepsilon^{-1})^d |X|$.*

Vapnik and Chervonenkis [28] described a probabilistic construction of ε-approximations in bounded VC-dimension. The deterministic construction stated above is due to Chazelle and Matoušek [8], and builds on earlier work [7, 17, 18, 21]. Haussler and Welzl [15] proved the upper bound on the size of ε-nets. The running time for computing an ε-net was improved to $O(d)^{3d}(\varepsilon^{-1} \log d\varepsilon^{-1})^d |X|$ by Brönnimann, Chazelle, and Matoušek [3], using the concept of a *sensitive ε-approximation*. Komlós, Pach, and Woeginger [16] showed that, for any fixed d, the bound of $O(\varepsilon^{-1} \log \varepsilon^{-1})$ for ε-nets is optimal in the worst case (see also [25]). The situation is different with ε-approximations: if $d > 1$ is the VC dimension, then there exists an ε-approximation for $(X, \mathcal{R})$ of size $O(\varepsilon^{-2+2/(d+1)})$ [22, 24].

An important application of ε-approximations is for estimating how many vertices in an arrangement of hyperplanes in $\mathbf{R}^d$ lie within a given convex region. Let $\Sigma = (H, \mathcal{R})$ be the set system formed by a set H of hyperplanes in $\mathbf{R}^d$, where each $R \in \mathcal{R}$ is the subset of H intersected by an arbitrary line segment. Let σ be a convex body (not necessarily full-dimensional). In the arrangement formed by H within the affine span of σ, let $V(H, \sigma)$ be the set of vertices that lie inside σ. The following was proven in [3, 4].

THEOREM 1.4 *Given a set H of hyperplanes in $\mathbf{R}^d$ in general position, let A be an ε-approximation for $\Sigma = (H, \mathcal{R})$. Given any convex body σ of dimension $k \leq d$,*

$$\left| \frac{|V(H,\sigma)|}{|H|^k} - \frac{|V(A,\sigma)|}{|A|^k} \right| \leq \varepsilon.$$

1.2.2 Optimal Cuttings

For convenience of exposition, we may assume that the set H of n hyperplanes in $\mathbf{R}^d$ is in general position. Let $\mathcal{A}(H)$ denote the arrangement formed by H. Obviously, no simplex of an ε-cutting can enclose more than $O(\varepsilon n)^d$ vertices. Since $\mathcal{A}(H)$ itself has exactly $\binom{n}{d}$ vertices, we should expect to need at least on the order of ε^{-d} simplices. But this is precisely the upper bound claimed in Theorem 1.1, which therefore is asymptotically tight.

Our starting point is an ε-net N for H, where the underlying set system $(X, \mathcal{R})$ is formed by a set X of hyperplanes and the collection $\mathcal{R}$ of subsets obtained by intersecting X with all possible open d-simplices. Its VC-dimension is bounded, and so by Theorem 1.3 an ε-net N of size $O(\varepsilon^{-1} \log \varepsilon^{-1})$ can be found in $n\varepsilon^{-O(1)}$ time.

We need to use a systematic way to triangulate the arrangement formed by the ε-net. We build a *canonical triangulation* of $\mathcal{A}(N)$ by induction on the dimension d (Fig. 1.2). The case $d = 1$ is trivial, so we assume that $d > 1$.

1. Rank the vertices of $\mathcal{A}(N)$ by the lexicographic order of their coordinate sequences.
2. By induction, form a canonical triangulation of the $(d-1)$-dimensional arrangement made by each hyperplane with respect to the $n-1$ others.
3. For each cell (ie, full-dimensional face) σ of $\mathcal{A}(N)$, lift toward its lowest-ranked vertex v each k-simplex ($k = 0, \ldots, d-2$) on the triangulated boundary of σ that does not lie in a $(d-1)$-face of $\mathcal{A}(N)$ that is incident to v.

It is not hard to see that the combinatorial complexity (ie, number of all faces of all dimensions) of the canonical triangulation of $\mathcal{A}(N)$ is asymptotically the same as that of $\mathcal{A}(N)$, which is $O(\varepsilon^{-1} \log \varepsilon^{-1})^d$. Therefore, the closures of its cells constitute an ε-cutting for H of size $O(\varepsilon^{-1} \log \varepsilon^{-1})^d$, which is good but not perfect. For optimality we must remove the log factor.

Assume that we have at our disposal an optimal method for building an ε_0-cutting of size $O(\varepsilon_0^{-d})$, for some suitably small constant ε_0. To bootstrap this into an optimal ε-cutting construction for any ε, we might proceed as follows: Beginning with a constant-size cutting, we progressively refine it by producing several generations of finer and finer cuttings, $\mathcal{C}_1, \mathcal{C}_2$, etc, where $\mathcal{C}_k$ is an ε_0^k-cutting for H of size $O(\varepsilon^{-dk})$. Specifically, assume that we have recursively computed the cutting $\mathcal{C}_k$ for H. For each $\sigma \in \mathcal{C}_k$, we have the incidence list H_σ of the hyperplanes intersecting the interior of σ. To compute the next-generation cutting $\mathcal{C}_{k+1}$, consider refining each σ in turn as follows:

1. Construct an ε_0-cutting for H_σ, using the algorithm whose existence is assumed.

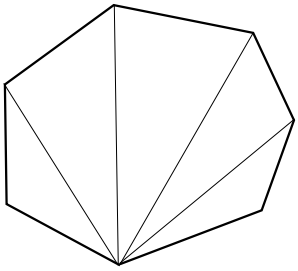

FIGURE 1.2: *A canonical triangulation.*

2. Retain only those simplices that intersect σ and clip them outside of σ.
3. In case the clipping produces nonsimplicial cells within σ, retriangulate them "canonically" (Fig. 1.3).

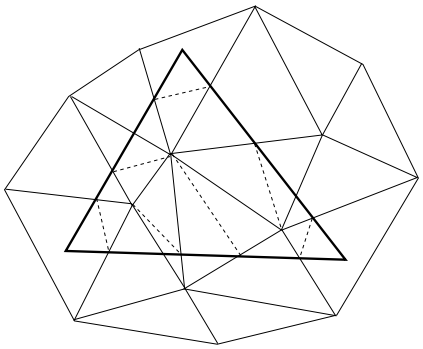

FIGURE 1.3: *Clip and retriangulate.*

Let $\mathcal{C}_{k+1}$ denote the collection of new simplices. A simplex of $\mathcal{C}_{k+1}$ in σ is cut (in its interior) by at most $\varepsilon_0|H_\sigma|$ hyperplanes of H_σ, and hence of H. By induction, this produces at most $n\varepsilon_0^{k+1}$ cuts; therefore, $\mathcal{C}_{k+1}$ is an ε_0^{k+1}-cutting. The only problem is that $\mathcal{C}_{k+1}$ might be a little too big. The reason is that excess in size builds up from generation to generation. We circumvent this difficulty by using a global parameter that is independent of the construction; namely, the total number of vertices.

Note that we may assume that $|H_\sigma| > n\varepsilon_0^{k+1}$, since σ would otherwise already satisfy the requirement of the next generation. We distinguish between *full* and *sparse* simplices. Given a set X of hyperplanes and a d-dimensional (closed) simplex σ, let $v(X, \sigma)$ be the number of vertices of $\mathcal{A}(X)$ in the interior of σ.

- The simplex $\sigma \in \mathcal{C}_k$ is *full* if $v(H, \sigma) \geq c_0|H_\sigma|^d$, where $c_0 = \varepsilon_0^2$. If so, we compute an ε_0-net for H_σ, and triangulate the portion of the net's arrangement within σ to form an ε_0-cutting of size $O(\varepsilon_0^{-1} \log \varepsilon_0^{-1})^d$. Its simplices form the elements of

$\mathcal{C}_{k+1}$ that lie within σ.

- A simplex σ that is not full is *sparse*. If so, we find a subset H_σ^o of H_σ that satisfies two conditions:
  - (i) The canonically triangulated portion of $\mathcal{A}(H_\sigma^o)$ that lies inside σ consists of a set $\mathcal{C}_\sigma^o$ of at most $\frac{1}{2}\varepsilon_0^{-d}$ full-dimensional (closed) simplices.
  - (ii) Each simplex of $\mathcal{C}_\sigma^o$ is intersected in its interior by at most $\varepsilon_0|H_\sigma|$ hyperplanes of H.

  The elements of $\mathcal{C}_{k+1}$ within σ are precisely the simplices of $\mathcal{C}_\sigma^o$.

LEMMA 1.3 $\mathcal{C}_{k+1}$ is an ε_0^{k+1}-cutting of size $O(\varepsilon_0^{-d(k+1)})$.

Next, we explain how to enforce conditions (i) and (ii) for sparse simplices. To be able to distinguish between full and sparse simplices, we use a $c_0/2$-approximation A_σ for H_σ of constant size, which we can build in $O(|H_\sigma|)$ time (Theorem 1.3). It follows from Theorem 1.4 that

$$\left| \frac{v(H,\sigma)}{|H_\sigma|^d} - \frac{v(A_\sigma,\sigma)|}{|A_\sigma|^d} \right| \leq \frac{c_0}{2}\,; \tag{1.1}$$

therefore, we can estimate $v(H,\sigma)$ in constant time with an error of at most $\frac{c_0}{2}|H_\sigma|^d$, which for our purposes here is inconsequential.

How do we go about refining σ and how costly is it? If σ is a full simplex, then by Theorem 1.3, we can compute the required ε_0-net in $O(|H_\sigma|)$ time. Within the same amount of time, we can also find the new set of simplices in σ, together with all of their incidence lists.

The refinement of a sparse simplex σ is a little more involved. We begin with a randomized construction, from which we then remove all the randomness. We compute H_σ^o by choosing a random sample from A_σ of size $c_1\varepsilon_0^{-1}\log\varepsilon_0^{-1}$, for some constant c_1 large enough (independent of ε_0). It can be shown that, with probability at least $2/3$, the sample forms an $(\varepsilon_0/2)$-net for A_σ. By Lemma 1.2, H_σ^o is a $(c_0/2+\varepsilon_0/2)$-net for H_σ; therefore, we ensure that (ii) holds with probability at least $2/3$. A slightly more complex analysis shows that (i) also holds with probability at least $2/3$; therefore (i,ii) are both true with probability at least $1/3$. We derandomize the construction in a trivial manner by trying out all possible samples, which takes constant time; therefore, the running time for refining σ is $O(|H_\sigma|)$.

Putting everything together, we see that refining any simplex takes time proportional to the total size of the incidence lists produced. By Lemma 1.3, the time needed for building generation $k+1$ is $O(n\varepsilon_0^{-(d-1)(k+1)})$. The construction goes on until we reach the first generation such that $\varepsilon_0^k \leq \varepsilon$. This establishes Theorem 1.1.

From the proof above it is not difficult to derive a rough estimate on the constant factor in the $O(\varepsilon^{-d})$ bound on the size of an ε-cutting. A thorough investigation into the smallest possible constant was undertaken by Har-Peled [14] for the two-dimensional case.

1.3 Applications

Cuttings have numerous uses in computational geometry. We mention just a handful: point location, Hopcroft's problem, convex hulls, Voronoi diagrams, and range searching. In many cases, cuttings allow us to derandomize existing probabilistic solutions, ie, to remove any

need for random bits and thus produce deterministic algorithms. Many other applications are described in the survey [2].

1.3.1 Point Location

How do we preprocess n hyperplanes in $\mathbf{R}^d$, so that, given a query point q, we can quickly find the face of the arrangement formed by the hyperplanes that contains the point? For an answer, simply set $\varepsilon = 1/n$ in Theorem 1.1, and use the nesting structure of $\mathcal{C}_1$, $\mathcal{C}_2$, etc, to locate q in $\mathcal{C}_k$. Note that this can be done in constant time once we know the location in $\mathcal{C}_{k-1}$.

THEOREM 1.5 *Point location among n hyperplanes can be done in $O(\log n)$ query time, using $O(n^d)$ preprocessing.*

Observe that if we only wish to determine whether the point q lies on one of the hyperplanes, it is possible to cut down the storage requirement a little. To do that, we use an ε-cutting for $\varepsilon = (\log n)/n$. The cells associated with the bottom of the hierarchy are each cut by $O(\log n)$ hyperplanes, which we can therefore check one by one. This reduces the storage to $O(n^d/(\log n)^{d-1})$.

1.3.2 Hopcroft's problem

Given n points and n lines in $\mathbf{R}^2$, is there any incidence between points and lines? This is *Hopcroft's problem.* It is self-dual; therefore dualizing it won't help. A classical arrangement of n lines due to Erdős has the property that its n highest-degree vertices are each incident to $\Omega(n^{1/3})$ edges. By picking these n lines as input to Hopcroft's problem and positioning the n points in the near vicinity of these high-degree vertices, we get a sense (not a proof) that to solve the problem should require checking each point against the $\Omega(n^{1/3})$ lines incident to their nearby vertex. This leads to an $\Omega(n^{4/3})$ running time, which under some realistic (though restrictive) conditions, can be made into a rigorous lower bound [13]. At the very least this line of reasoning suggests that to beat $\Omega(n^{4/3})$ is unlikely to be easy. This bound has almost been achieved by an algorithm of Matoušek [20] with, at its heart, a highly intricate and subtle use of cuttings.

THEOREM 1.6 *To decide whether n points and n lines in the plane are free of any incidence can be done in $n^{4/3}\, 2^{O(\log^* n)}$ time.*

1.3.3 Convex Hulls and Voronoi Diagrams

Cuttings play a key role in computing convex hulls in higher dimension. Given n points in $\mathbf{R}^d$, their convex hull is a bounded convex polytope with $O(n^{\lfloor d/2 \rfloor})$ vertices. Of course, it may have much fewer of them: eg, $d+1$, if $n-d-1$ points lie strictly inside the convex hull of the $d+1$ others. It is notoriously difficult to design *output-sensitive* algorithms, the term designating algorithms whose running time is a function of both input and output sizes. In the "worst case" approach our goal is a simpler one: to design an optimal convex hull agorithm that runs in $O(n \log n + n^{\lfloor d/2 \rfloor})$ time. (The extra term $n \log n$ is unavoidable because sorting is easily embedded as a convex hull problem.)

Computing the convex hull of n points is equivalent by duality to computing the intersection of n halfspaces. A naive approach to this problem is to insert each halfspace one after

the other while maintaining the intersection of previously inserted halfspaces incrementally. This can be done without difficulty if we maintain a canonical triangulation of the current intersection polyhedron and update a bipartite graph indicating which hyperplane intersects which cell of the triangulation. A surprising fact, first proven by Clarkson and Shor [11], is that if the halfspaces are inserted in random order, then the expected running time of the algorithm can be made optimal. By using an elaborate mix of ε-nets, ε-approximations, and ε-cuttings, Chazelle [5] showed how to compute the intersection deterministically in optimal time; his algorithm was subsequently simplified by Brönnimann, Chazelle, and Matoušek [3]; a complete description is also given in the book [6]. This implies the two theorems below.

THEOREM 1.7 *The polyhedron formed by the intersection of n halfspaces in $\mathbf{R}^d$ can be computed in $O(n \log n + n^{\lfloor d/2 \rfloor})$ time.*

Not only does this result give us an optimal deterministic solution for convex hulls, but it also solves the Voronoi diagram problem. Indeed, recall [12, 29] that a Voronoi diagram of n points in $\mathbf{R}^d$ can be "read off" from the facial structure of the convex hull of a lift of the n points into $\mathbf{R}^{d+1}$.

THEOREM 1.8 *The convex hull of a set of n points in $\mathbf{R}^d$ can be computed deterministically in $O(n \log n + n^{\lfloor d/2 \rfloor})$ time. By duality, the Voronoi diagram (or Delaunay triangulation) of a set of n points in $\mathbf{E}^d$ can be computed deterministically in $O(n \log n + n^{\lceil d/2 \rceil})$ time.*

1.3.4 Range Searching

Simplex range searching refers to the problem of preprocessing a set P of n points in $\mathbf{R}^d$ so that, given a query (closed) simplex σ, the size of $P \cap \sigma$ can be quickly evaluated. Variants of the problem include reporting the points of $P \cap \sigma$ explicitly or, assuming that each point p has a weight $w(p) \in \mathbf{R}$, computing $\sum \{ w(p) \mid p \in P \cap \sigma \}$. The most powerful data structure for solving simplex range searching, the *simplicial partition*, vividly illustrates the power of ε-cuttings. A collection $\{(P_i, R_i)\}$ is called a *simplicial partition* if

- the collection $\{P_i\}$ forms a partition of P; and
- each R_i is a relatively open simplex that contains P_i.

The simplices R_i can be of any dimension and, in fact, need not even be disjoint; furthermore the P_i's need not be equal to $P \cap R_i$. A hyperplane is said to *cut* R_i if it intersects, but does not contain, R_i. The *cutting number* of the simplicial partition refers to the maximum number of R_i's that can be cut by a single hyperplane. Matoušek [19] designed an optimal construction, which happens to be crucially based on ε-cuttings.

LEMMA 1.4 Given a set P of n points in $\mathbf{R}^d$ ($d > 1$), for any integer $1 < r \leq n/2$, there exists a simplicial partition of cutting number $O(r^{1-1/d})$ such that $n/r \leq |P_i| < 2n/r$ for each (P_i, R_i) in the partition.

To understand the usefulness of simplicial partitions for range searching, one needs to learn about *partition trees*. A partition tree for P is a tree $\mathcal{T}$ whose root is associated with the point set P. The set P is partitioned into subsets $P_1, \ldots, P_m$, with each P_i associated

with a distinct child v_i of the root. To each v_i corresponds a convex open set R_i, called the *region* of v_i, that contains P_i. The regions R_i are not necessarily disjoint. If $|P_i| > 1$, the subtree rooted at v_i is defined recursively with respect to P_i.

Armed with a partition tree, it is a simple matter to handle range search queries. In preprocessing, at each node we store the sum of the weights of the points associated with the corresponding region. To answer a query σ, we visit all the children v_i of the root and check whether σ intersects the region R_i of v_i: (i) if the answer is yes, but σ does not completely enclose the region R_i of v_i, then we visit v_i and recurse; (ii) if the answer is yes, but σ completely encloses R_i, we add to our current weight count the sum of the weights within P_i, which happens to be stored at v_i; (iii) if the answer is no, then we do not recurse at v_i.

It is straightforward to see that Lemma 1.4 can be used to construct partition trees. It remains for us to choose the branching factor. If we choose a large enough constant r, we end up with a partition tree that lets us answer simplex range search queries in $O(n^{1-1/d+\varepsilon})$ time for any fixed $\varepsilon > 0$, using only $O(n)$ storage. A more complex argument by Matoušek [19] removes the ε term from the exponent.

With superlinear storage, various space-time tradeoffs can be achieved. For example, as shown by Chazelle, Sharir, and Welzl [9], simplex range searching with respect to n points in $\mathbf{R}^d$ can be done in $O(n^{1+\varepsilon}/m^{1/d})$ query time, using a data structure of size m, for any $n \leq m \leq n^d$. Matoušek [20] slightly improved the query time to $O(n(\log m/n)^{d+1}/m^{1/d})$, for m/n large enough. These bounds are essentially optimal under highly general computational models [6].

Acknowledgments

This work was supported in part by NSF grant CCR-998817, and ARO Grant DAAH04-96-1-0181.

References

[1] Agarwal, P.K. Partitioning arrangements of lines II: Applications, *Disc. Comput. Geom.* 5 (1990), 533–573.

[2] Agarwal, P.K. Geometric partitioning and its applications, in *Computational Geometry: Papers from the DIMACS Special Year,* eds., Goodman, J.E., Pollack, R., Steiger, W., Amer. Math. Soc., 1991.

[3] Brönnimann, H., Chazelle, B., Matoušek, J. Product range spaces, sensitive sampling, and derandomization, *SIAM J. Comput.* 28 (1999), 1552–1575.

[4] Chazelle, B. Cutting hyperplanes for divide-and-conquer, *Disc. Comput. Geom.* 9 (1993), 145–158.

[5] Chazelle, B. An optimal convex hull algorithm in any fixed dimension, *Disc. Comput. Geom.* 10 (1993), 377–409.

[6] Chazelle, B. *The Discrepancy Method: Randomness and Complexity*, Cambridge University Press, hardcover 2000, paperback 2001.

[7] Chazelle, B., Friedman, J. A deterministic view of random sampling and its use in geometry, *Combinatorica* 10 (1990), 229–249.

[8] Chazelle, B., Matoušek, J. On linear-time deterministic algorithms for optimization problems in fixed dimension, *J. Algorithms* 21 (1996), 579–597.

[9] Chazelle, B., Sharir, M., Welzl, E. Quasi-optimal upper bounds for simplex range searching and new zone theorems, *Algorithmica* 8 (1992), 407–429.

[10] Clarkson, K.L. New applications of random sampling in computational geometry, *Disc. Comput. Geom.* 2 (1987), 195–222.
[11] Clarkson, K.L., Shor, P.W. Applications of random sampling in computational geometry, II, *Disc. Comput. Geom.* 4 (1989), 387–421.
[12] Edelsbrunner, H. *Algorithms in Combinatorial Geometry*, Springer, 1987.
[13] Erickson, J. New lower bounds for Hopcroft's problem, *Disc. Comput. Geom.* 16 (1996), 389–418.
[14] Har-Peled, S. Constructing planar cuttings in theory and practice, *SIAM J. Comput.* 29 (2000), 2016–2039.
[15] Haussler, D., Welzl, E. ε-nets and simplex range queries, *Disc. Comput. Geom.* 2 (1987), 127–151.
[16] Komlós, J., Pach, J., Woeginger, G. Almost tight bounds for ε-nets, *Disc. Comput. Geom.* 7 (1992), 163–173.
[17] Matoušek, J. Construction of ε-nets, *Disc. Comput. Geom.* 5 (1990), 427–448.
[18] Matoušek, J. Cutting hyperplane arrangements, *Disc. Comput. Geom.* 6 (1991), 385–406.
[19] Matoušek, J. Efficient partition trees, *Disc. Comput. Geom.* 8 (1992), 315–334.
[20] Matoušek, J. Range searching with efficient hierarchical cuttings, *Disc. Comput. Geom.* 10 (1993), 157–182.
[21] Matoušek, J. Approximations and optimal geometric divide-and-conquer, *J. Comput. Syst. Sci.* 50 (1995), 203–208.
[22] Matoušek, J. Tight upper bounds for the discrepancy of halfspaces, *Disc. Comput. Geom.* 13 (1995), 593–601.
[23] Matoušek, J. *Geometric Discrepancy: An Illustrated Guide*, Algorithms and Combinatorics, 18, Springer, 1999.
[24] Matoušek, J., Welzl, E., Wernisch, L. Discrepancy and ε-approximations for bounded VC-dimension, *Combinatorica* 13 (1993), 455–466.
[25] Pach, J., Agarwal, P.K. *Combinatorial Geometry*, Wiley-Interscience Series in Discrete Mathematics and Optimization, John Wiley & Sons, Inc., 1995.
[26] Sauer, N. On the density of families of sets, *J. Combinatorial Theory A* 13 (1972), 145–147.
[27] Shelah, S. A combinatorial problem; stability and order for models and theories in infinitary languages, *Pac. J. Math.* 41 (1972), 247–261.
[28] Vapnik, V.N., Chervonenkis, A.Ya. On the uniform convergence of relative frequencies of events to their probabilities, *Theory of Probability and its Applications* 16 (1971), 264–280.
[29] Ziegler, G.M. *Lectures on Polytopes*, Graduate Texts in Mathematics, 152, Springer-Verlag, 1995.

Index