SIAM J. COMPUT.
Vol. 40, No. 4, pp. 1045–1074

SEMIALGEBRAIC RANGE REPORTING AND EMPTINESS SEARCHING WITH APPLICATIONS*

MICHA SHARIR† AND HAYIM SHAUL‡

Abstract. In a typical range-emptiness searching (resp., reporting) problem, we are given a set P of n points in $\mathbb{R}^d$, and we wish to preprocess it into a data structure that supports efficient range-emptiness (resp., reporting) queries, in which we specify a range σ, which, in general, is a semialgebraic set in $\mathbb{R}^d$ of constant description complexity, and we wish to determine whether $P \cap \sigma = \emptyset$, or to report all the points in $P \cap \sigma$. Range-emptiness searching and reporting arise in many applications and have been treated by Matoušek [*Comput. Geom. Theory Appl.*, 2 (1992), pp. 169–186] in the special case where the ranges are half-spaces bounded by hyperplanes. As shown in Matoušek's work, the two problems are closely related, and they have solutions (for the case of half-spaces) with similar performance bounds. In this paper we extend the analysis to general semialgebraic ranges and show how to adapt Matoušek's technique without the need to *linearize* the ranges into a higher-dimensional space. This yields more efficient solutions to several useful problems, and we demonstrate the new technique in four applications with the following results: (i) An algorithm for ray shooting amid balls in $\mathbb{R}^3$, which uses $O(n)$ storage and $O^*(n)$ preprocessing (we use the notation $O^*(n^\gamma)$ to mean an upper bound of the form $C(\varepsilon)n^{\gamma+\varepsilon}$, which holds for any $\varepsilon > 0$, where $C(\varepsilon)$ is a constant that depends on ε) and answers a query in $O^*(n^{2/3})$ time, improving the previous bound of $O^*(n^{3/4})$. (ii) An algorithm that preprocesses, in $O^*(n)$ time, a set P of n points in $\mathbb{R}^3$ into a data structure with $O(n)$ storage, so that, for any query line ℓ (or, for that matter, any simply shaped convex set), the point of P farthest from ℓ can be computed in $O^*(n^{1/2})$ time. This in turn yields an algorithm that computes the largest-area triangle spanned by P in time $O^*(n^{26/11})$, as well as nontrivial algorithms for computing the largest-perimeter or largest-height triangle spanned by P. (iii) An algorithm that preprocesses, in $O^*(n)$ time, a set P of n points in $\mathbb{R}^2$ into a data structure with $O(n)$ storage, so that, for any query α-fat triangle Δ, we can determine, in $O^*(1)$ time, whether $\Delta \cap P$ is empty. Alternatively, we can report, in $O^*(1) + O(k)$ time, the points of $\Delta \cap P$, where $k = |\Delta \cap P|$. (iv) An algorithm that preprocesses, in $O^*(n)$ time, a set P of n points in $\mathbb{R}^2$ into a data structure with $O(n)$ storage, so that, given any query semidisk c, or a circular cap larger than a semidisk, we can determine, in $O^*(1)$ time, whether $c \cap P$ is empty, or report the k points in $c \cap P$ in $O^*(1) + O(k)$ time. Adapting the recent techniques of [B. Aronov and S. Har-Peled, *SIAM J. Comput.*, 38 (2008), pp. 899–921, B. Aronov, S. Har-Peled, and M. Sharir, *On approximate halfspace range counting and relative epsilon-approximations*, in Proceedings of the 23rd ACM Symposium Comput. Geom., 2007, pp. 327–336, B. Aronov and M. Sharir, *SIAM J. Comput.*, 39 (2010), pp. 2704–2725], we can turn our solutions into efficient algorithms for approximate range counting (with small relative error) for the cases mentioned above. Our technique is closely related to the notions of nearest- or farthest-neighbor generalized Voronoi diagrams and of the union or intersection of geometric objects, where sharper bounds on the combinatorial complexity of (decompositions of complements of) these structures yield faster range-emptiness searching or reporting algorithms.

Key words. range searching, semialgebraic sets, range emptiness, range reporting, random sampling, elementary cell partition, epsilon nets, ray shooting

AMS subject classifications. 68P05, 68P10, 68Q25, 68Q87, 68W20, 68W40

DOI. 10.1137/090765092

*Received by the editors July 15, 2009; accepted for publication (in revised form) April 5, 2011; published electronically July 19, 2011. This work was supported by grants 155/05 and 338/09 from the Israel Science Fund, by NSF grants CCF-05-14079 and CCF-08-30272, by grant 2006-194 from the U.S.-Israel Binational Science Foundation, and by the Hermann Minkowski–MINERVA Center for Geometry at Tel Aviv University. This work is part of the second author's Ph.D. dissertation, prepared under the supervision of the first author at Tel Aviv University.

http://www.siam.org/journals/sicomp/40-4/76509.html

†School of Computer Science, Tel Aviv University, Tel Aviv 69978, Israel and Courant Institute of Mathematical Sciences, New York University, New York, NY 10012 (michas@post.tau.ac.il).

‡School of Computer Science, Tel Aviv University, Tel Aviv 69978, Israel (hayim@post.tau.ac.il).

1. Introduction. The main technical contribution of this paper is an extension of Matoušek's range-emptiness and reporting data structures [30] (see also [4] for a dynamic version of the problem) to the case of general semialgebraic ranges.

Ray shooting amid balls. A motivating application of this study is ray shooting amid balls in $\mathbb{R}^3$, where we want to construct a data structure of linear size with near-linear preprocessing, which supports ray shooting queries in sublinear time. Typically, in problems of this sort, the bound on the query time is some fractional power of n, the number of objects, and the goal is to make the exponent as small as possible. For example, ray shooting amid a collection of n arbitrary triangles can be performed in $O^*(n^{3/4})$ time (with linear storage) [3]. Better solutions are known for various special cases. For example, the authors have shown [37] that the query time can be improved to $O^*(n^{2/3})$ when the triangles are all *fat* or are all stabbed by a common line.

At the other end of the spectrum, one is interested in ray-shooting algorithms and data structures where a ray-shooting query can be performed in logarithmic or polylogarithmic time (or even $O(n^\varepsilon)$ time for any $\varepsilon > 0$; this is $O^*(1)$ in our shorthand notation). In this case, the goal is to reduce the storage (and preprocessing) requirements as much as possible. For example, for arbitrary triangles (and even for the special case of fat triangles), the best known bound for the storage requirement (with logarithmic query time) is $O^*(n^4)$ [1, 3]. For balls, Mohaban and Sharir [34], gave an algorithm with $O^*(n^3)$ storage and $O^*(1)$ query time. However, when only linear storage is used, the previously best known query time (for balls) is $O^*(n^{3/4})$ (as in the case of general triangles). In this paper we show, as an application of our general range-emptiness machinery, that this can be improved to $O^*(n^{2/3})$ time.

When answering a ray-shooting query for a set S of input objects, one generally reduces the problem to that of answering *segment-emptiness* queries, following the parametric-searching scheme proposed by Agarwal and Matoušek [2] (see also [33] for the original underlying technique).

A standard way of performing the latter kind of queries is to switch to a dual parametric space, where each object in the input set is represented by a *point*. A segment e in $\mathbb{R}^3$ is mapped to a surface σ_e, which is the locus of all the points representing the objects that e touches (without penetrating into their interior). Usually, σ_e partitions the dual space into two portions, one, σ_e^+, consisting of points representing objects whose interior is intersected by e, and the other, σ_e^-, consisting of points representing objects that e avoids. The segment-emptiness problem thus transforms into a range-emptiness query: Does σ_e^+ contain any point representing an input object?

Range reporting and emptiness searching. Range-emptiness queries of this kind have been studied by Matoušek [30] (see also [4]), but only for the case where the ranges are half-spaces bounded by hyperplanes. For this case, Matoušek has established a so-called *shallow-cutting lemma* that shows the existence of a $(1/s)$-cutting[1] that covers the complement of the union of any m given half-space ranges, whose size is significantly smaller than the size of a $(1/s)$-cutting that covers the entire space. This lemma provides the basic tool for partitioning a point set P, in the style of [31] so that *shallow* hyperplanes (those containing at most n/r points of P below them, say, for some given parameter r) cross only a small number of cells

[1] This is a partition of space (or a portion thereof) into a small number of simply shaped cells, each of which is crossed by at most n/s of the n given surfaces (hyperplanes in this case). See below for more details.

of the partition (see below for more details). This in turn yields a data structure, known as a *shallow partition tree*, that stores a recursive partitioning of P, which enables us to answer more efficiently half-space range *reporting* queries for shallow hyperplanes, and thus also half-space range-emptiness queries. Using this approach, the query time (for emptiness) improves from the general half-space range searching query cost of $O^*(n^{1-1/d})$ to $O^*(n^{1-1/\lfloor d/2\rfloor})$. Reporting (for arbitrary hyperplanes) takes $O^*(n^{1-1/\lfloor d/2\rfloor}+k)$, where k is the output size.

Consequently, one way of applying this machinery for more general semialgebraic ranges is to "lift" the set of points and the ranges into a higher-dimensional space by means of an appropriate *linearization*, as in [3], and then apply the above machinery. (For this, one needs to assume that the given ranges have *constant description complexity*, meaning that each range is a Boolean combination of a constant number of polynomial equalities and inequalities of constant maximum degree.) However, if the space in which the ranges are linearized has high dimension, the resulting range reporting or emptiness queries become significantly less efficient. Moreover, in many applications, the ranges are Boolean combinations of several polynomial (equalities and) inequalities, which creates additional difficulties in linearizing the ranges, resulting in even worse running time.

An alternative technique is to give up linearization and instead work in the original space. As follows from the machinery of [30] (and further elaborated later in this paper), this requires, as a major tool, the (existence and) construction of a decomposition of the *complement of the union* of m given ranges (in the case of segment emptiness, these are the ranges σ_e^+, for an appropriate collection of segments e), into a small number of "elementary cells" (in the terminology of [3]—see also below). Here we face, especially in higher dimensions, a scarcity of sharp bounds on the complexity of the union itself, to begin with, and then on the complexity of a decomposition of its complement. Often, the best one can do is to decompose the entire arrangement of the given ranges, which results in too many elementary cells and consequently in an algorithm with poor performance.

To recap, in the key technical step in answering general semialgebraic range reporting or emptiness queries, the best current approaches are either to construct a cutting of the *entire* arrangement of the range-bounding surfaces in the original space, or to construct a shallow cutting in another higher-dimensional space into which the ranges can be linearized. For many natural problems (including the segment-emptiness problem), both approaches yield relatively poor performance.

As we will shortly note, in handling general semialgebraic ranges, we face another major technical issue, having to do with the construction of efficient *test sets* of ranges (in the terminology of [3], elaborated below). Addressing this issue is a major component of the analysis in this paper, and is discussed in detail later on.

Our results. We propose a variant of the shallow-cutting machinery of [30] for the case of semialgebraic ranges, which avoids the need for linearization, and works in the original space (which, for the case of ray shooting amid balls, is a 4-dimensional parametric space in which the balls are represented as points). While the machinery used by our variant is similar in principle to that in [30], there are several significant technical difficulties which require more careful treatment.

Matoušek's technique [30], as well as ours, considers a finite set Q of shallow ranges (called a *test set*), and builds a data structure which caters only for ranges in Q. Matoušek shows how to build, for any given parameter r, a set of half-spaces of size polynomial in r, which represents well *all* (n/r)-shallow ranges in the following sense: For any *simplicial partition*[2] Π with parameter r, let κ denote the maximal number of cells of Π crossed by a half-space in Q. Then each (n/r)-shallow half-space crosses at most $c\kappa$ cells of Π, where c is a constant that depends on the dimension; here a half-space crosses a cell if it intersects the cell but does not fully contain it. Unfortunately (for the present analysis), the linear nature of the ranges is crucially needed for the proof, which therefore fails for nonlinear ranges.

Being a good representative of all shallow ranges, in the above sense, is only one of the requirements from a good test set Q. The other requirements are that Q be small so that, in particular, it can be constructed efficiently, and that the (decomposition of the complement of) the union of any subset of Q have small complexity. All these properties hold for the case of half-spaces bounded by hyperplanes, studied in [30].

As it turns out, and as was hinted above, obtaining a "good" test set Q for general semialgebraic ranges with the above properties is not an easy task. We give a simple general recipe for constructing such a set Q, but it consists of more complex ranges than those in the original setup. A major problem with this recipe is that since the members of Q have a more complex shape, it becomes harder to establish good bounds on the complexity of (the decomposition of the complement of the) union of any subset of these generalized ranges.

Nevertheless, once a good test set has been shown to exist, and to be efficiently computable, it leads to a construction of an efficient elementary-cell partition with a small crossing number for any empty or shallow original range. Using this construction recursively, one obtains a *partition tree*, of linear size, so that any shallow original range γ visits only a small number of its nodes (where γ visits a node if it *crosses* the elementary cell enclosing the subset of that node, meaning, as above, that it intersects this cell but does not fully contain it), which in turn leads to an efficient range reporting or emptiness-testing procedure. This part of constructing and searching the tree is almost identical to its counterparts in the earlier works [3, 30, 31], and we will not elaborate on it here, focusing only on the technicalities in the construction of a single "shallow" elementary-cell partition.

Developing all this machinery, and then putting it into action, we obtain efficient data structures for the following applications, improving previous results or obtaining the first nontrivial solutions. These instances follow.

Ray shooting amid balls in 3*-space.* Given a set S of n, possibly intersecting, balls in $\mathbb{R}^3$, we construct, in $O^*(n)$ time, a data structure of $O(n)$ size, which can determine, for a given query segment e, whether e is empty (avoids all balls) in $O^*(n^{2/3})$ time. Plugging this data structure into the parametric-searching technique of Agarwal and Matoušek [2], we obtain a data structure for answering ray-shooting queries amid the balls of S, which has similar performance bounds.

We represent balls in 3-space as points in $\mathbb{R}^4$, where a ball with center (a, b, c) and radius r is mapped to the point (a, b, c, r), and each object $K \subset \mathbb{R}^3$ is mapped to the surface σ_K, which is the locus of all (points representing) balls tangent to K (i.e., balls that touch K, but do not penetrate into its interior). In this case, the range of an object K is the upper half-space σ_K^+ consisting of all points lying above σ_K (representing balls that intersect K). The complement of the union of a subfamily of these ranges is the region below the lower envelope of the corresponding surfaces[3] σ_K.

[2]Briefly, this is a partition of P into $O(r)$ subsets of roughly equal size, each enclosed by some simplex (in the linear case) or some elementary cell (in the general semialgebraic case); see [3] and section 3 below.

[3]In our solution, we will use a test set of objects K which are considerably more complex than just lines or segments, but are nevertheless still of constant description complexity.

The *minimization diagram* of this envelope is the 3-dimensional Euclidean Voronoi diagram of the corresponding set of objects. Thus we reveal (what we regard as) a somewhat surprising connection between the problem of ray shooting amid balls and the problem of analyzing the complexity of Euclidean Voronoi diagrams of (simply shaped) objects in 3-space.

Farthest point from a line (or from any convex set) in $\mathbb{R}^3$. Let P be a set of n points in $\mathbb{R}^3$. We wish to preprocess P into a data structure of size $O(n)$, so that, for any query line ℓ, we can efficiently find the point of P farthest from ℓ. This is a useful routine for approximating polygonal paths in three dimensions; see [17].

As in the ray-shooting problem, we can reduce such a query to a range-emptiness query of the form: Given a cylinder C, does it contain all the points of P? (That is, is the complement of the cylinder empty?) We prefer to regard this as an instance of the complementary *range-fullness* problem, which seeks to determine whether a query range is *full* (i.e., contains all the input points).

Our machinery can handle this problem. In fact, we can solve the range-fullness problem for any family of *convex* ranges in 3-space, of constant description complexity. Our solution requires $O(n)$ storage and near-linear preprocessing, and it answers a range-fullness query in $O^*(n^{1/2})$ time, improving the query time $O^*(n^{2/3})$ given by Agarwal and Matoušek [3].

We then apply this result to solve the problem of finding the largest-area triangle spanned by a set of n points in 3-space. The resulting algorithm requires $O^*(n^{26/11})$ time, which improves a previous bound of $O^*(n^{13/5})$ due to Daescu and Serfling [17]. We also adapt our machinery to compute efficiently the largest-perimeter triangle and the largest-height triangle spanned by such a point set.

In both this, and the preceding ray-shooting applications, we use the general, more abstract recipe for constructing good test sets.

Fat triangle and circular cap range-emptiness searching and reporting. Finally, we consider two planar instances of the range-emptiness and reporting problems, in which we are given a planar set P of n points, and the ranges are either *α-fat triangles* or sufficiently large *circular caps* (say, larger than a semidisk). The general technique of Agarwal and Matoušek [3] yields, for any class of planar ranges with constant description complexity, a data structure with near-linear preprocessing and linear storage, which answers such queries in time $O^*(n^{1/2})$ (for emptiness) or $O^*(n^{1/2}) + O(k)$ (for reporting). We improve the query time to $O^*(1)$ and $O^*(1) + O(k)$, respectively, in both cases. (The case of fat triangles has also an alternative known efficient solution; see section 6 for details.)

In these planar applications, we abandon the general recipe and construct good test sets in an ad-hoc (and simpler) manner. For α-fat triangles (i.e., triangles with the property that each of their angles is at least α, which is some fixed positive constant), the test set consists of "canonical" $(\alpha/2)$-fat triangles, and the fast query performance is a consequence of the fact that the complexity of the complement of the union of m α'-fat triangles is $O(m \log^* m)$ for any constant $\alpha' > 0$ [9](see also [23, 32]). It is quite likely that our machinery can also be applied to other classes of fat objects in the plane for which near-linear bounds on the complexity of their union are known [18, 19, 20, 21]. However, constructing a good test set for each of these classes is not an obvious step. We leave these extensions as open problems for further research.

For circular caps, the motivation for range-emptiness searching comes from the problem of finding, for a query consisting of a point q and a line ℓ, the point of P which lies above ℓ and is nearest to q (we only consider the case where q lies on or above ℓ). Such a procedure was considered in [16]. Using parametric searching, the latter problem can be reduced to that of testing for emptiness of a circular cap centered at q and bounded by ℓ (the assumption on the location of q ensures that this cap is at least a semidisk). Here too we manage to construct a test set which consists of (possibly slightly smaller) circular caps, and we exploit the fact that the complexity of the union of m such caps is $O^*(m)$, as long as the caps are not too small (relative to their bounding circles), to obtain the fast performance stated above.

Approximate range counting. Adapting the recent techniques of [10, 11, 12], we can turn our solutions into efficient algorithms for approximate range counting (with small relative error) for the cases mentioned above. That is, for a specified $\delta > 0$, we can preprocess the input point set P into a data structure which can efficiently compute, for any query range γ, an approximate count t_γ, satisfying $(1-\delta)|P \cap \gamma| \leq t_\gamma \leq (1+\delta)|P \cap \gamma|$. The performance of the resulting algorithms is detailed in section 7. As observed in the papers just cited, approximate range counting is closely related to the range-emptiness problem, which in fact is a special case of the former problem. The algorithm in [10] performs approximate range counting by a randomized binary search over $|P \cap \gamma|$, where the search is guided by repeated calls to an emptiness-testing routine on various random samples of P. This algorithm uses emptiness searching as a black box, so, plugging our solutions for this latter problem into their algorithm, we obtain efficient approximate range counting algorithms for the ranges considered in this paper. See section 7 for details.

Related work. Our study was originally motivated by work by Daescu and others [16, 17] on path approximations and related problems. In these applications one needs to compute efficiently the vertex of a subpath which is farthest from a given segment (connecting the two endpoints of the subpath). These works used the standard range searching machinery of [3], and motivated us to look for faster implementations.

The general range-emptiness (or reporting) problem was studied by the authors a few years ago [38]. In this earlier version, we did not manage to handle properly the issue of constructing a good test set, so the results presented there are somewhat incomplete. The present paper builds upon the previous one, but it provides a thorough analysis of this aspect of the problem and consequently obtains a complete and efficient solution to the problems listed above, and it lays down the foundation for obtaining efficient solutions to many other similar problems—we believe indeed that the applications given here only scratch the surface of the wealth of potential future applications of this sort.

2. Preliminaries and notations. We begin with a brief review of the main concepts and notations used in our analysis.

Range spaces. A range space is a pair (X, Γ), where X is a set and $\Gamma \subseteq 2^X$ is a collection of subsets of X, called *ranges*. In our applications, $X = \mathbb{R}^d$, and Γ is a collection of semialgebraic sets of some specific type, each having *constant description complexity*. That is, each set in Γ is given as a Boolean combination of a constant number of polynomial equalities and inequalities of constant maximum degree. To simplify the analysis, we assume,[4] as in [3], that all the ranges in Γ are defined by a single Boolean combination so that each polynomial p in this combination is $(d+t)$-variate, and each range γ has t degrees of freedom so that if we substitute the values of these t parameters into the last t variables of each p, the resulting Boolean combination defines the range γ. This allows us to represent the ranges of Γ as points in an appropriate t-dimensional parametric space.

[4]This assumption is not essential and is only made to simplify the presentation.

Under these special assumptions, the range space (X, Γ) has *finite VC-dimension*, a property formally defined in [25]. Informally, it ensures that, for any finite subset P of X, the number of distinct ranges of P is $O(|P|^\delta)$, where δ is the VC-dimension.

As a matter of fact, we will consider range spaces of the form (P, Γ_P), where $P \subset \mathbb{R}^d$ is a finite point set, and each range in Γ_P is the intersection of P with a range in Γ.

Cuttings. Given a finite collection Γ of n semialgebraic ranges in $\mathbb{R}^d$, as above, and a parameter $r < n$, a $(1/r)$-*cutting* for Γ is a partition Ξ of $\mathbb{R}^d$ (or of some portion of $\mathbb{R}^d$) into a finite number of relatively open cells of dimensions $0, 1, \ldots, d$ so that each cell is *crossed* by at most n/r ranges of Γ, where a range $\gamma \in \Gamma$ is said to cross a cell σ if $\gamma \cap \sigma \neq \emptyset$, but γ does not fully contain σ. We will also need to consider *weighted* $(1/r)$-cuttings, where each range $\gamma \in \Gamma$ has a positive weight $w(\gamma)$, and each cell of Ξ is crossed by ranges whose total weight is at most W/r, where $W = \sum_{\gamma \in \Gamma} w(\gamma)$ is the overall weight of all the ranges in Γ.

Shallow ranges. A range $\gamma \in \Gamma$ is called k-*shallow* with respect to a set P of points in $\mathbb{R}^d$ if $|\gamma \cap P| \leq k$.

Elementary cells. Define, as in [3], an *elementary cell* in $\mathbb{R}^d$ to be a connected relatively open semialgebraic set of some dimension $k \leq d$, which is homeomorphic to a ball and has constant description complexity. As above, we assume, for simplicity, that the elementary cells are defined by a single Boolean combination involving t free variables, and each cell is determined by fixing the values of these t parameters.

Elementary-cell partition. Let P be a set of n points in $\mathbb{R}^d$. An *elementary-cell partition* of P is a collection $\Pi = \{(P_1, s_1), \ldots, (P_m, s_m)\}$, for some integer m, such that (i) $\{P_1, \ldots, P_m\}$ is a partition of P (into pairwise disjoint subsets), and (ii) each s_i is an elementary cell that contains the respective subset P_i. In general, the cells s_i need not be disjoint. Usually, one also specifies a parameter $r \leq n$ and requires that $n/r \leq |P_i| \leq 2n/r$ for each i, so $m = O(r)$.

The function $\zeta(r)$. In Lemma 3.1 and Theorem 3.2, we use a function $\zeta(r)$ that bounds the number of elementary cells in a decomposition of the complement of the union of any r ranges of Γ. We assume that $\zeta(r)$ is "well behaved" in the sense that for each $c > 0$ there exists $c' > 0$ such that $\zeta(cr) \leq c'\zeta(r)$ for every r. We also assume that $\zeta(r) = \Omega(r)$.

(ν, α)-*samples and shallow* ε-*nets.* We recall the result of Li, Long, and Srinivasan [29], and we adapt it, similar to the recent observations in [26], to obtain a useful extension of the notion of ε-nets. Both assumptions are natural and hold practically in all applications.

Let $(X, \mathcal{R})$ be a range space of finite VC-dimension δ, and let $0 < \alpha, \nu < 1$ be two given parameters. Consider the distance function

$$d_\nu(r, s) = \frac{|r - s|}{r + s + \nu} \qquad \text{for } r, s \geq 0.$$

A subset $N \subseteq X$ is called a (ν, α)-*sample* if for each $R \in \mathcal{R}$ we have

$$d_\nu\left(\frac{|X \cap R|}{|X|}, \frac{|N \cap R|}{|N|}\right) < \alpha.$$

THEOREM 2.1 (Li, Long, and Srinivasan [29]). *A random sample N of*

$$O\left(\frac{1}{\alpha^2 \nu}\left(\delta \log \frac{1}{\nu} + \log \frac{1}{q}\right)\right)$$

elements of X is a (ν, α)-sample with probability at least $1 - q$ (with an appropriate choice of the constant of proportionality).

Har-Peled and Sharir [26] show that, by appropriately choosing α and ν, various standard constructs, such as ε-nets and ε-approximations, are special cases of (ν, α)-samples. Here we follow a similar approach, and we show the existence of small-size *shallow ε-nets*, a new notation introduced in this paper.

Let us first define this notion. Let $(X, \mathcal{R})$ be a range space of finite VC-dimension δ, and let $0 < \varepsilon < 1$ be a given parameter. A subset $N \subseteq X$ is a shallow ε-net if it satisfies the following two properties, for some absolute constant c.

(i) For each $R \in \mathcal{R}$ and for any parameter $t \geq 0$, if $|N \cap R| \leq t \log \frac{1}{\varepsilon}$, then $|X \cap R| \leq c(t+1)\varepsilon|X|$.

(ii) For each $R \in \mathcal{R}$ and for any parameter $t \geq 0$, if $|X \cap R| \leq t\varepsilon|X|$, then $|N \cap R| \leq c(t+1) \log \frac{1}{\varepsilon}$.

Note the difference between shallow and standard ε-nets. Property (i) (with $t = 0$) implies that a shallow ε-net is also a standard ε-net (possibly with a recalibration of ε). Property (ii) has no parallel in the case of standard ε-nets—there is no guarantee how a standard net interacts with small ranges.

THEOREM 2.2. *A random sample N of*

$$O\left(\frac{1}{\varepsilon}\left(\delta \log \frac{1}{\varepsilon} + \log \frac{1}{q}\right)\right)$$

elements of X is a shallow ε-net with probability at least $1 - q$ (again, with an appropriate choice of the constant of proportionality).

Proof. Take $\alpha = 1/2$, say, and calibrate the constants in the size of N to guarantee with probability $1 - q$ that N is an $(\varepsilon, 1/2)$-sample. Assume that this is indeed the case. For a range $R \in \mathcal{R}$, put $X_R = |X \cap R|/|X|$ and $N_R = |N \cap R|/|N|$. We have

$$d_\varepsilon(X_R, N_R) = \frac{|X_R - N_R|}{X_R + N_R + \varepsilon} < \frac{1}{2}.$$

That is, $|X_R - N_R| < \frac{1}{2}(X_R + N_R + \varepsilon)$, or $X_R < 3N_R + \varepsilon$, and, symmetrically, $N_R < 3X_R + \varepsilon$. This is easily seen to imply properties (i) and (ii). For (i), let R be a range for which $|N \cap R| \leq t \log \frac{1}{\varepsilon}$; that is, $N_R \leq \beta t \varepsilon$ for some absolute constant β (proportional to the VC-dimension). Then

$$|X \cap R| = |X| \cdot X_R < |X|(3N_R + \varepsilon) \leq (3\beta t + 1)\varepsilon|X|.$$

For (ii), let R be a range for which $|X \cap R| \leq t\varepsilon|X|$; that is, $X_R \leq t\varepsilon$. Then

$$|N \cap R| = |N| \cdot N_R < |N|(3X_R + \varepsilon) \leq (3t + 1)\varepsilon|N| \leq (3t + 1)\gamma \log \frac{1}{\varepsilon}$$

for another absolute constant γ (again, proportional to the VC-dimension). ∎

3. Semialgebraic range reporting and emptiness searching.

3.1. Shallow cutting in the semialgebraic case. We begin by extending the shallow-cutting lemma of Matoušek [30] to the more general setting of semialgebraic ranges. This extension is fairly straightforward, although it involves several technical steps that deserve to be highlighted. The proof of this lemma is given in Appendix A. Similar to the general cutting lemma, Lemma 3.1 also holds for the weighted case, where each range of Γ is associated with a weight, and where each cell of the $(1/r)$-cutting is crossed by ranges whose total weight is at most $1/r$ of the overall weight of the ranges of Γ.

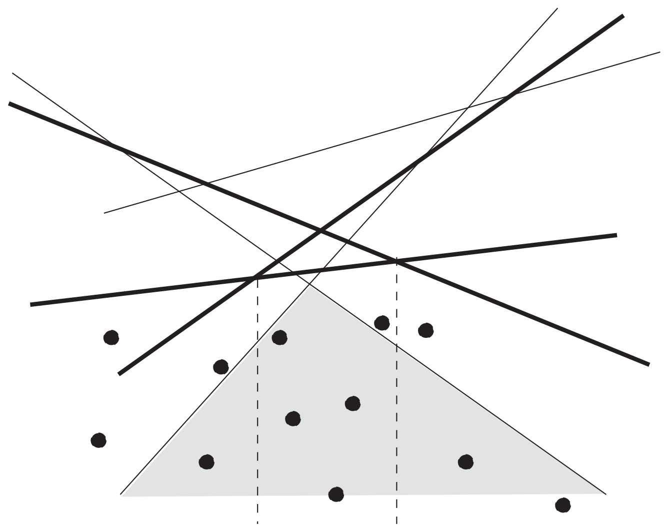

FIG. 1. *A planar point set and a collection Γ of upper half-planes. A random sample of the lines bounding these ranges is shown in bold with a decomposition of the region below their lower envelope, which contains the region below the lower envelope of all the bounding lines, drawn shaded.*

LEMMA 3.1 (extended shallow-cutting lemma). *Let Γ be a collection of n semi-algebraic ranges in $\mathbb{R}^d$. Assume that the complement of the union of any subset of m ranges in Γ can be decomposed into at most $\zeta(m)$ elementary cells for a well-behaved function ζ as above. Then, for any $r \le n$, there exists a $(1/r)$-cutting Ξ of Γ with the following properties:*

(i) *The union of the cells of Ξ contains the complement of the union of Γ.*

(ii) *Ξ consists of $O(\zeta(r))$ elementary cells.*

(iii) *The complement of the union of the cells of Ξ is contained in the union of $O(r)$ ranges in Γ.*

A special case that frequently arises is one in which each range in Γ is an upper half-space bounded by the graph of some continuous $(d-1)$-variate function. In this case the complement K of the union of r ranges is the portion of space that lies below the *lower envelope* of the bounding graphs. In this case, it suffices to decompose the graph of the lower envelope itself into at most $\zeta(r)$ elementary cells. Indeed, having done that, we can extend each cell τ within the envelope into the cell τ^- consisting of all points that lie vertically below τ. The new cells decompose K and are also elementary. See Figure 1 for an illustration. Symmetric properties hold for the case of lower half-spaces.

As already discussed in the introduction, obtaining tight or nearly tight bounds for $\zeta(r)$ is still a major open problem for many instances of the above setup. For example, decomposing an upper envelope of r $(d-1)$-variate functions of constant description complexity into $O^*(r^{d-1})$ elementary cells is still open for any $d \ge 4$. (This bound is the best possible in the worst case, since it is the worst-case tight bound on the complexity of such an undecomposed envelope [36].) The cases $d = 2$ (upper envelope of curves in the plane) and $d = 3$ (upper envelope of 2-dimensional surfaces in 3-space) are easy. In these cases $\zeta(r)$ is proportional to the complexity of the envelope, which in the worst case is near-linear for $d = 2$ and near-quadratic for $d = 3$ [36]. In higher dimensions, the only general-purpose bound known to date is the upper bound obtained by computing the *vertical decomposition* of the *entire* arrangement of the given surfaces and extracting from it the relevant cells that lie on or above the envelope. In particular, for $d = 4$ the bound is $\zeta(r) = O^*(r^4)$, as follows from the results of [28]. This leaves a gap of about a factor of r between this bound and the bound $O^*(r^3)$ on the complexity of the undecomposed envelope. For arbitrary $d \geq 4$, the bound is $\zeta(r) = O^*(r^{2d-4})$ with progressively increasing gaps from the lower bound $\Omega^*(r^{d-1})$. Of course, in certain special cases, most notably the case of hyperplanes, as studied in [30], both the envelope and its decomposition have (considerably) smaller complexity.

The situation with the complexity of the union of geometric objects is even worse. While considerable progress was recently made on many special cases in two and three dimensions (see [6] for a recent comprehensive survey), there are only very few sharp bounds on the complexity of unions in higher dimensions. Worse still, even when a sharp bound on the complexity of the union is known, obtaining comparable bounds on the complexity of a decomposition of the complement of the union is a much harder problem (in $d \geq 3$ dimensions). As an example, the union of n infinite cylinders in 3-space is known to have near-quadratic complexity [7, 22], but it is still an open problem whether its complement can be decomposed into a near-quadratic number of elementary cells.

3.2. Partition theorem for shallow semialgebraic ranges. We next apply the new shallow-cutting lemma to construct an elementary cell partition of a given input point set P with respect to a specific set Q of ranges. This is done in a fairly similar way to that in [3] (see also [30, 31]). A major difference in handling the semialgebraic case is the construction of a set Q of ranges (called a *test set*, as in [3, 30, 31]) that will be (a) small enough, and (b) representative of all shallow (or empty) ranges, in a sense discussed in detail below. The method given in [3] does not work in the general semialgebraic case, and different, sometimes ad hoc, approaches need to be taken.

The following theorem summarizes the main part of the construction (except for the construction of Q).

THEOREM 3.2 (extended partition theorem). *Let P be a set of n points in $\mathbb{R}^d$, let Γ be a family of semialgebraic ranges of constant description complexity, and let r be fixed. Let Q be another finite collection (not necessarily a subset of Γ) of semialgebraic ranges of constant description complexity with the following properties:* (i) *The ranges in Q are all (n/r)-shallow.* (ii) *The complement of the union of any m ranges of Q can be decomposed into at most $\zeta(m)$ elementary cells for any m.* (iii) *Any (n/r)-shallow range $\gamma \in \Gamma$ can be covered by the union of at most δ ranges of Q, where δ is a constant.*

Then there exists an elementary cell partition Π of P, of size $O(r)$, into subsets of size roughly n/r such that the crossing number of any (n/r)-shallow range in Γ with the cells of Π is either $O(r/\zeta^{-1}(r) + \log r \log |Q|)$ if there exists a fixed $\varepsilon > 0$ such that $\zeta(r)/r^{1+\varepsilon}$ is monotone increasing, or $O(r \log r/\zeta^{-1}(r) + \log r \log |Q|)$ otherwise.

The proof, which, again, is similar to those in [3, 30, 31], proceeds through the following steps. We first have the following lemma.

LEMMA 3.3. *Let P be a set of n points in $\mathbb{R}^d$, and let $r < n$ be a parameter. Let Q be a set of (n/r)-shallow ranges with the property that the complement of the union of any subset of m ranges of Q can be decomposed into at most $\zeta(m)$ elementary cells for any m. Then there exists a subset $P' \subseteq P$ of at least $n/2$ points and an elementary cell partition $\Pi = \{(P_1, s_1), \ldots, (P_m, s_m)\}$ for P' with $n/r < |P_i| < 2n/r$ for all i such that each range of Q crosses at most $O(r/\zeta^{-1}(r) + \log |Q|)$ cells s_i of Π.*

Proof. We will inductively construct disjoint sets $P_1, \ldots, P_m \subset P$ of size n/r and elementary cells $s_1, \ldots, s_m$ such that $P_i \subseteq s_i$ for each i. The construction terminates when $|P_1 \cup \cdots \cup P_m| \geq n/2$. Suppose that $P_1, \ldots, P_{i-1}$ have already been constructed, and set $P'_i := P \setminus \bigcup_{j<i} P_j$. We construct P_i as follows. For a range $\sigma \in Q$, let $\kappa_i(\sigma)$ denote the number of cells among $s_1, \ldots, s_{i-1}$ crossed by σ. We define a weighted collection (Q, w_i) of ranges so that each range $\sigma \in Q$ appears with weight (or multiplicity) $w_i(\sigma) = 2^{\kappa_i(\sigma)}$. We put $w_i(Q) = \sum_{\sigma \in Q} w_i(\sigma)$. By Lemma 3.1 and by our assumption that the function $\zeta(r)$ is well behaved, there exists a $(1/t)$-cutting Ξ_i for the weighted collection (Q, w_i) of size at most $r/4$, for an appropriate choice of $t = \Theta(\zeta^{-1}(r))$, with the following properties: the union of Ξ_i contains the complement of the union of Q, and the complement of the union of Ξ_i is contained in the union of $O(t)$ ranges of Q. Since all these ranges are (n/r)-shallow, the number of points of P not in the union of Ξ_i is at most $O(t) \cdot (n/r) = n \cdot O(\zeta^{-1}(r)/r)$, and our assumptions on $\zeta(r)$ imply that this is smaller than $n/4$ if we choose t with an appropriate constant of proportionality. Since we assume that $|P'_i| \geq n/2$, it follows that at least $n/4$ points of P'_i lie in the union of the at most $r/4$ cells of Ξ_i. By the pigeonhole principle, there is a cell s_i of Ξ_i containing at least n/r points of P'_i. We take P_i to be some subset of $P'_i \cap s_i$ of size exactly n/r, and we make s_i the cell in the partition which contains P_i.

We next establish the asserted bound on the crossing numbers between the ranges of Q and the elementary cells $s_1, \ldots, s_m$ in the following standard manner. The final weight $w_m(\sigma)$ of a range $\sigma \in Q$ with crossing number $\kappa = \kappa(\sigma)$ (with respect to the final partition) is 2^κ. On the other hand, each newly added cell s_i is crossed by ranges of Q of total weight $O(w_i(Q)/\zeta^{-1}(r))$, because s_i is an elementary cell of the corresponding weighted $(1/t)$-cutting Ξ_i. The weight of each of these crossing ranges is doubled at the ith step, and the weight of all the other ranges remains unchanged. Thus $w_{i+1}(Q) \leq w_i(Q)(1 + O(1/\zeta^{-1}(r)))$. Hence, for each range $\sigma \in Q$ we have

$$\begin{aligned} w_m(\sigma) \leq w_m(Q) \leq |Q| \left(1 + O\left(\frac{1}{\zeta^{-1}(r)}\right)\right)^m \leq |Q| \left(1 + O\left(\frac{1}{\zeta^{-1}(r)}\right)\right)^{O(r)} \\ \leq |Q| e^{O(r/\zeta^{-1}(r))}, \end{aligned}$$

and thus $\kappa = \log w_m(\sigma) = O(r/\zeta^{-1}(r) + \log |Q|)$. ∎

Discussion. The limitation of Lemma 3.3 is that the bound that it derives (a) applies only to ranges in Q, and (b) includes the term $\log |Q|$. An ingenious component of the analysis in [30] overcomes both problems by choosing a test set Q of ranges whose size is only polynomial in r (and, in particular, is independent of n), which is nevertheless sufficiently representative of all shallow ranges in the sense that the crossing number of any (n/r)-shallow range is $O(\max\{\kappa(\sigma) \mid \sigma \in Q\})$. This implies that Lemma 3.3 holds for *all* shallow ranges with the stronger bound which does not involve $\log |Q|$.

Unfortunately, the technique of [30] does not extend to the case of semialgebraic ranges, as it crucially relies on the linearity of the ranges.[5] The following lemma gives a sufficient condition for a test set Q to be representative of the relevant shallow ranges in the sense that Q satisfies the assumptions made in Theorem 3.2.

[5]It uses point-hyperplane duality and exploits the fact that a half-space (bounded by a hyperplane) intersects a simplex if and only if it contains a vertex of the simplex, which is false in the general semialgebraic case.

LEMMA 3.4. *Let P be a set of n points in $\mathbb{R}^d$, and let Γ be a family of semialgebraic ranges with constant description complexity. Consider an elementary-cell partition $\Pi = \{(P_1, s_1), \ldots, (P_r, s_r)\}$ of P such that $n/r < |P_i| < 2n/r$ for each i. Let Q be a finite set of (n/r)-shallow ranges (not necessarily ranges of Γ) so that the maximal crossing number of a range $q \in Q$ with respect to Π is κ. Then, for any range $\gamma \in \Gamma$ which is contained in the union of at most δ ranges of Q (for some constant δ), the crossing number of γ is at most $(\kappa+1)\delta$.*

Proof. Let $\gamma \in \Gamma$ be a range for which there exist δ ranges $q_1, \ldots, q_\delta$ of Q such that $\gamma \subseteq q_1 \cup \cdots \cup q_\delta$. Then, if γ crosses a cell s_i of Π, at least one of the covering ranges q_j must either cross s_i or fully contain s_i. The number of cells of Π that can be crossed by any single q_j is at most κ, and each q_j can fully contain at most one cell of Π (because q_j is (n/r)-shallow).[6] Hence, the overall number of cells of Π that γ can cross is at most $(\kappa+1)\delta$, as asserted. □

Proof of Theorem 3.2. Apply Lemma 3.3 to the input set $P_0 = P$ with parameter $r_0 = r$. This yields an elementary-cell partition Π_0 for (at least) half of the points of P_0, which satisfies the properties of that lemma. Let P_1 denote the set of the remaining points of P_0, and set $r_1 = r_0/2$. Apply Lemma 3.3 again to P_1 with parameter r_1, obtaining an elementary-cell partition Π_1 for (at least) half of the points of P_1. We iterate this process $k = O(\log r)$ times until the set P_k has fewer than $2n/r$ points. We take Π to be the union of all the elementary-cell partitions Π_i formed so far, together with one large cell containing all the remaining points of P_k. The resulting elementary-cell partition of P consists of at most $1 + r + r/2 + r/4 + \cdots \leq 2r$ subsets, each of size at most $2n/r$. The crossing number of a range in Q is, by Lemma 3.3,

$$O\left(\sum_{i=1}^{\log r} \left((r/2^i)/\zeta^{-1}(r/2^i) + \log|Q|\right)\right).$$

Our assumptions on ζ imply that if there exists an $\varepsilon > 0$ such that $\zeta(r)/r^{1+\varepsilon}$ is monotone increasing, then the first terms add up to $O(r/\zeta^{-1}(r))$; otherwise we can bound their sum by $O(r \log r/\zeta^{-1}(r))$. Hence, by the properties of Q and by Lemma 3.4, the crossing number of any empty range is also $O(r/\zeta^{-1}(r) + \log|Q| \log r)$ or $O(r \log r/\zeta^{-1}(r) + \log|Q| \log r)$, respectively.

Remark. With some care, the term $O(\log r \log|Q|)$ can be reduced to $O(\log|Q|)$ by following the arguments in the proofs in [31] or [30]. Of course, either version of this term is dominated by the other term in all our applications. □

3.3. Partition trees for reporting and emptiness searching. As in the classical works on range searching [3, 30, 31], we choose r to be a sufficiently large constant and apply Theorem 3.2 recursively to obtain a *partition tree* $\mathcal{T}$, where each node v of $\mathcal{T}$ stores a subset P_v of P and an elementary cell σ_v enclosing P_v. The children of a node v are obtained from an elementary-cell partition of P_v—each of them stores one of the resulting subsets of P_v and its enclosing cell. The size of the subset that is stored at a leaf is $O(r)$. As in previous works, the partition tree requires linear storage.

Testing a range γ for emptiness is done by searching with γ in $\mathcal{T}$. At each visited node v, where $\gamma \cap \sigma_v \neq \emptyset$, we test whether $\gamma \supseteq \sigma_v$, in which case γ is not empty. Otherwise, we find the children of v whose cells are intersected by γ. If there are too

[6] By choosing a slightly smaller value for r in the construction of the partition, we can even rule out the possibility that a range q_j fully contains a cell of Π. This, however, has no effect on the asymptotic bounds that the analysis derives.

many of them we know that γ is (not shallow and thus) not empty. Otherwise, we recurse at each child.

Reporting is performed in a similar manner. If $\sigma_v \subseteq \gamma$, we output all of σ_v. Otherwise, we find the children of v whose cells are intersected by γ. If there are too many of them, we know that γ is not (n_v/r)-shallow (with respect to P_v), so, if r is a constant, we can afford to check every element of P_v for containment in γ and output those points that do lie in γ. If there are not too many children, we recurse in each of them.

The efficiency of the search depends on the function $\zeta(m)$. If $\zeta(m) = O^*(m^k)$, then an emptiness query takes $O^*(n^{1-1/k})$ time, and a reporting query takes $O^*(n^{1-1/k}) + O(t)$, where t is the output size. Thus making ζ (i.e., k) small is the main challenge in this technique.

3.4. A general recipe for constructing good test sets. Let Γ be the given collection of semialgebraic ranges of constant description complexity. As above, we assume that each range $\gamma \in \Gamma$ has t degrees of freedom for some constant parameter t so it can be represented as a point γ^* in a t-dimensional parametric space, which, for convenience, we denote as $\mathbb{R}^t$. Each input point $p \in P$ is mapped to a region K_p, which is the locus of all points representing ranges that contain p.

We fix a parameter $r \geq 1$, and we choose a random sample N of $ar \log r$ points of P, where a is a sufficiently large constant. We form the set $N^* = \{K_p \mid p \in N\}$, construct the arrangement $\mathcal{A}(N^*)$, and let $V = A_{\leq k}(N^*)$ denote the region consisting of all points contained in at most k ranges of N^*, where $k = b \log r$ and b is an absolute constant that we will fix later. We decompose V into elementary cells, using, e.g., vertical decomposition [36]. In the worst case, we get $O^*(r^{2t-4})$ elementary cells [14, 28].[7]

Let τ be one of these cells. We associate with τ a generalized range γ_τ in $\mathbb{R}^d$, which is the union $\bigcup\{\gamma \mid \gamma^* \in \tau\}$. Since τ has constant description complexity, as do the ranges of Γ, it is easy to show that γ_τ is also a semialgebraic set of constant description complexity (see [13]).

We define the test set Q to consist of all the generalized ranges γ_τ, over all cells τ in the decomposition of V, and we claim that, with high probability (and with an appropriate choice of b), Q is a good test set in the following three aspects.

(i) *Compactness.* $|Q| = O^*(r^{2t-4})$; that is, the size of Q is polynomial in r and independent of n.

(ii) *Shallowness.* Each range γ_τ in Q is $\beta(n/r)$-shallow with respect to P for some constant parameter β.

(iii) *Containment.* Every (n/r)-*shallow* range $\gamma \in \Gamma$ is contained in a single range γ_τ of Q.

Property (i) is obvious. For (ii), consider the range space (P, Γ^*), where Γ^* consists of all generalized ranges γ_τ over all elementary cells τ of the form arising in the above vertical decomposition. It is a fairly easy exercise to show that (P, Γ^*) also has finite VC-dimension. See, e.g., [36]. By Theorem 2.2, if a is a sufficiently large constant (proportional to the VC-dimension of (P, Γ^*)), then N is a shallow $(1/r)$-net for both range spaces (P, Γ) and (P, Γ^*), with high probability, so we assume that N is indeed such a shallow $(1/r)$-net.

Let $\gamma_\tau \in Q$. Note that any point $p \in P$ in γ_τ lies in a range $\gamma \in \Gamma$ with $\gamma^* \in \tau$. By definition, γ^* also belongs to K_p, and so K_p crosses or fully contains τ. Since

[7]Here, in this dual construction, we do not need any sharper bound; any bound polynomial in r is sufficient for our purpose.

τ is $(b\log r)$-shallow in $\mathcal{A}(N^*)$, it is fully contained in at most $b\log r$ regions K_p for $p \in N$ (and is not crossed by any such region). Hence, $|\gamma_\tau \cap N| < b\log r$, so, since N is a shallow $(1/r)$-net for (P, Γ^*), we have $|\gamma_\tau \cap P| < c(b+1)n/r$, so γ_τ is $(c(b+1)n/r)$-shallow, which establishes (ii).

For (iii), let $\gamma \in \Gamma$ be an (n/r)-shallow range. Since N is a shallow $(1/r)$-net for (P, Γ), and $|\gamma \cap P| \le |P|/r$, we have $|\gamma \cap N| \le 2c\log r$ for some constant $c > 0$. Hence, with $b \ge 2c$, $\gamma \in V$, so there is a cell τ of the decomposition which contains γ, which by construction implies that $\gamma \subseteq \gamma_\tau$, thus establishing (iii).

To make Q a really good test set, we also need the following fourth property.

(iv) *Efficiency.* There exists a good bound on the associated function $\zeta(m)$, bounding the size of a decomposition of the complement of the union of any m ranges of Q.

The potentially rather complex shape of these generalized ranges makes it harder to obtain, in general, a good bound on ζ.

In what follows we manage to use this general recipe in two of our four applications (ray shooting amid balls and range-fullness searching) with good bounds on the corresponding functions $\zeta(\cdot)$. In two other planar applications (range-emptiness searching with fat triangles and with circular caps), we abandon the general technique, and we construct ad hoc good test sets.

Remark. In the preceding construction, we wanted to make sure that every (n/r)-shallow range $\gamma \in \Gamma$ is covered by a range of Q. If we only need this property for *empty* ranges γ (which is the case for emptiness testing), it suffices to consider only the 0-level of $\mathcal{A}(N^*)$, i.e., the complement of the union of N^*. Other than this simplification, the construction proceeds as above.

4. Fullness searching and reporting outliers for convex ranges. Let P be a set of n points in 3-space, and let Γ be a set of convex ranges of constant description complexity. We wish to preprocess P in near-linear time into a data structure of linear size so that, given a query range $\gamma \in \Gamma$, we can efficiently determine whether γ contains all the points of P. Alternatively, we want to report all the points of P that lie outside γ. This is clearly a special case of range-emptiness searching or range reporting if one considers the complements of the ranges in Γ. For simplicity, we mostly focus on the range-fullness problem; the extension to reporting "outliers" is similar to the standard treatment of reporting queries, as discussed earlier, and will be briefly addressed below.

We present a solution to this problem with $O^*(n^{1/2})$ query time, thereby improving over the best known general bound of $O^*(n^{2/3})$, given in [3], which applies to any range searching (e.g., range counting) with semialgebraic sets (of constant description complexity) in $\mathbb{R}^3$.

To apply our technique to this problem we first need to build a good test set. We apply the general recipe of section 3.4 to the complements of the ranges of Γ. With an appropriate use of De Morgan laws, we can express the construction in terms of the ranges of Γ themselves. Specifically, we take a random sample N as above, construct the *intersection* $I = \bigcap_{p\in N} K_p$, and decompose it into elementary cells. For each resulting cell σ, let γ_σ denote the *intersection* $\bigcap_{\gamma^*\in\sigma} \gamma$. As above, since σ has constant description complexity, γ_σ is a semialgebraic set of constant description complexity. Note that, since the ranges in Γ are convex, each range γ_σ is also convex (albeit of potentially more complex shape than that of the original ranges).

As in section 3.4, we take the test set Q to consist of all the generalized ranges γ_σ over all cells σ in the decomposition of I. We note that the ranges γ_σ are in fact the complements of the ranges that we would get if we applied the construction of section 3.4 to the complements of the ranges of Γ.

We argue that Q satisfies all four properties required from a good test set. Indeed, compactness, shallowness (or, rather, "almost fullness"), and containment follow from the arguments in section 3.4 (recalling that we work with complements of ranges). We therefore consider efficiency and show that the complexity of a decomposition of the intersection of any m ranges in Q (the complement of the union of m range complements) is $O^*(m^2)$, so $\zeta(m) = O^*(m^2)$.

CLAIM 4.1. *Let Q be a set of convex "almost full" ranges, each containing at least $n - n/r$ points of P. The intersection K of any m ranges $q_1, \ldots, q_m \in Q$ can be decomposed into $O^*(m^2)$ elementary cells.*

Proof. Since all ranges in Q are convex, K is a convex set too. Assume, for simplicity of presentation, that K is nonempty and has nonempty interior, and fix a point o in that interior. We can regard the boundary of each q_i as the graph of a bivariate function $\rho = F_i(\theta, \varphi)$ in spherical coordinates about o. Then ∂K is the graph of the lower envelopes of these functions. Since the q_i's have constant description complexity, (the graph of) each F_i is also a semialgebraic set of constant description complexity.[8] Hence the combinatorial complexity of ∂K is $O^*(m^2)$ [36]. Moreover, since ∂K is 2-dimensional, we can partition it into $O^*(m^2)$ trapezoidal-like elementary cells, using a variant of the vertical decomposition technique, and then extend each such cell τ_0 to a 3-dimensional cone-like cell τ, which is the union of all segments connecting o to the points of τ_0. The resulting cells τ constitute a decomposition of K into $O^*(m^2)$ elementary cells, as claimed. $\square$

Using the machinery developed in the preceding section, we therefore obtain the following result.

THEOREM 4.2. *Let P be a set of n points in $\mathbb{R}^3$, and let Γ be a family of convex ranges of constant description complexity. Then one can construct, in near-linear time, a data structure of linear size so that, for any range $\gamma \in \Gamma$, it can determine in $O^*(n^{1/2})$ time whether γ is full.*

Reporting outliers. To extend the above approach to the problem of reporting outliers, we apply a construction similar to that in the "general recipe" presented above. That is, we take the $b \log r$ deepest levels of $\mathcal{A}(N)$ for an appropriate constant b, decompose them into elementary cells, and construct a generalized range γ_σ for each of these cells σ. The general machinery given above implies the following result.

THEOREM 4.3. *Let P be a set of n points in $\mathbb{R}^3$, and let Γ be a family of convex ranges of constant description complexity. Then one can construct in near-linear time a data structure of linear size so that, for any range $\gamma \in \Gamma$, it can report the points of P in the complement of γ in $O^*(n^{1/2}) + O(k)$ time, where k is the query output size.*

4.1. Farthest point from a convex shape. A useful application of the data structure of Theorem 4.2 is to *farthest-point queries*. In such a problem we are given a set P of n points in $\mathbb{R}^3$, and we wish to preprocess it in near-linear time into a data structure of linear size so that, given a convex query object o (from some fixed class of objects with constant description complexity), we can efficiently find the point of P farthest from o.

We solve this problem using parametric searching [33]. The corresponding decision problem is: Given the query object o and a distance ρ, determine whether the Minkowski sum $o \oplus B_\rho$ is full, where B_ρ is the ball of radius ρ centered at the

[8] With an appropriate algebraic reparametrization of the spherical coordinates, of course.

origin. The smallest ρ with this property is the distance to the farthest point from o. With an appropriate small-depth parallel implementation of this decision problem, the parametric searching also takes time $O^*(n^{1/2})$. Reporting the k farthest points from o, for any parameter k, can be done in $O^*(n^{1/2}) + O(k)$ time, using a simple variant of this technique.

4.2. Computing the largest-area, largest-perimeter, and largest-height triangles. Let P be a set of n points in $\mathbb{R}^d$. We wish to find the triangle whose vertices belong to P and whose area (resp., perimeter, height) is maximal. This problem is a useful subroutine in path approximation algorithms; see [17]. Daescu and Serfling [17] gave an $O^*(n^{13/5})$-algorithm for the 3-dimensional largest-area triangle. In d dimensions, the running time is $O^*(n^{3-2/(\lfloor d^2/2 \rfloor+1)})$.

In $\mathbb{R}^3$, our technique, without any additional enhancements, already yields the improved bound $O^*(n^{5/2})$, using the following straightforward procedure. For each pair of points $p_1, p_2 \in P$, we find the farthest point $q \in P$ from the line $\overline{p_1p_2}$, compute the area of Δp_1p_2q, and output the largest-area triangle among those triangles. The procedure performs farthest-point queries from $O(n^2)$ lines for a total cost of $O^*(n^{5/2})$, as claimed.

We can improve this solution, using the following standard decomposition technique, to an algorithm with running time $O^*(n^{26/11})$. First, the approach just described performs M farthest-point queries on a set of N points in time $O^*(MN^{1/2} + N)$, where the second term is the preprocessing cost of preparing the data structure.

Before continuing, we note the following technical issue. Recall that we find the farthest point from a query line ℓ by drawing a cylinder C_ρ around ℓ, whose radius ρ is the smallest (unknown) radius for which C_ρ contains P. The concrete value of ρ is found using parametric searching. In the approach that we follow now, we will execute in parallel $O(n^2)$ different queries, each with its own ρ, so care has to be taken when running the parametric search with this multitude of different unknown values of ρ.

While there are several alternative solutions to this problem, we use the following one, which seems the cleanest. Let $A > 0$ be a fixed parameter. For each pair p_1, p_2 of distinct points of P, let $C_A(p_1p_2)$ denote the cylinder whose axis passes through p_1 and p_2 and whose radius is $2A/|p_1p_2|$. In the decision procedure, we specify the value of A and perform $O(n^2)$ range-fullness queries with the cylinders $C_A(p_1p_2)$. If all of them are found to be full, then $A \geq A^*$, where A^* is the (unknown) maximal area of a triangle spanned by P; otherwise $A < A^*$. (With a somewhat finer implementation, we can also distinguish between the cases $A > A^*$ and $A = A^*$; we omit the details of this refinement.)

To implement the decision procedure, we apply a duality transform, where each cylinder C in 3-space is mapped to a point $C^* = (a, b, c, d, \rho)$, where (a, b, c, d) is some parametrization of the axis of C and ρ is its radius. In this dual parametric 5-space, a point $p \in \mathbb{R}^3$ is mapped to a surface p^*, which is the locus of all (points representing) cylinders which contain p on their boundary. Note that the portion of space below (resp., above) p^* in the ρ-direction consists of points dual to cylinders which do not contain (resp., contain) p.

Let $P^* = \{p^* \mid p \in P\}$. Fix some sufficiently large but constant parameter r_0, and construct a $(1/r_0)$-cutting Ξ of $\mathcal{A}(P^*)$, using the vertical decomposition of a random sample of $O(r_0 \log r_0)$ surfaces of P^* (see, e.g., [36]). As follows from [14, 28], the combinatorial complexity of Ξ is $O^*(r_0^6)$. We distribute the $O(n^2)$ points dual to the query cylinders among the cells of Ξ, in brute force, and we also find, in equally brute force, for each cell τ the subset P^*_τ of surfaces which cross τ. We ignore cells which fully lie below some surface of P^*, because cylinders whose dual points fall in such a cell cannot be full (the decision algorithm stops as soon as such a point (cylinder) is detected). For each of the remaining cells τ, we repeat this procedure with the subset of the points dual to the surfaces in P^*_τ and with the subset of cylinders whose dual points lie in τ. We keep iterating in this manner until we reach cuttings whose cells are crossed by at most n/r dual surfaces, where r is some (nonconstant) parameter that we will shortly fix. As is easily checked, the overall number of cells in these cuttings is $O^*(r^6)$.

We then run the preceding weaker procedure on each of the resulting cells τ with the set P_τ of points dual to the surfaces which cross τ and with the set $\mathcal{C}_\tau$ of cylinders whose dual points lie in τ. Letting m_τ denote the number of these cylinders, the overall cost of the second phase of the procedure is

$$\sum_\tau O^*(m_\tau (n/r)^{1/2} + n/r) = O^*(n^2(n/r)^{1/2} + nr^5).$$

Since r_0 is a constant, the overall cost of the first phase is easily seen to be proportional to the overall size of the resulting subproblems, which is $O^*(n^2 + nr^5)$. Overall, the cost is thus

$$O^*(n^{5/2}/r^{1/2} + nr^5).$$

Choosing $r = n^{3/11}$, this becomes $O^*(n^{26/11})$.

Running a generic version of this decision procedure in parallel is fairly straightforward. The cuttings themselves depend only on the dual surfaces, which do not depend on A^*, so we can construct them in a concrete, nonparametric fashion. Locating the points dual to the query cylinders can be done in parallel, and, since r_0 is a constant, this takes constant parallel depth for each of the logarithmically many levels of cuttings. The second phase can also be executed in parallel in an obvious manner. Omitting the further easy details, we conclude that the overall algorithm also takes $O^*(n^{26/11})$ time.

Largest-perimeter triangle. The above technique can be adapted to yield efficient solutions of several problems of a similar flavor. For example, consider the problem of computing the largest-perimeter triangle among those spanned by a set P of n points in $\mathbb{R}^3$. Here, for each pair p_1, p_2 of points of P, and for a specified perimeter π, we construct the ellipsoid of revolution $E_\pi(p_1, p_2)$, whose boundary is the locus of all points q satisfying $|qp_1| + |qp_2| = \pi - |p_1p_2|$. (Here, of course, we only consider pairs p_1, p_2 with $|p_1p_2| < \pi/2$.) We now run $O(n^2)$ range-fullness queries with these ellipsoids, and we report that $\pi^* > \pi$ if at least one of these ellipsoids in not full, or $\pi^* \le \pi$ otherwise, where π^* is the largest perimeter. (Here too one can discriminate between $\pi^* < \pi$ and $\pi^* = \pi$; we omit the details as above.)

The efficient implementation of this procedure is carried out in a way similar to the preceding algorithm, except that here the dual representation of our ellipsoids require 6 degrees of freedom to specify the foci p_1 and p_2. Unlike the previous case, the dual surfaces p^* do depend on π, so in the generic implementation of the decision procedure we also need to construct the various $(1/r_0)$-cuttings in a generic, parallel manner.[9] However, since r_0 is a constant, this is easy to do in constant parallel depth per cutting. A $(1/r_0)$-cutting in $\mathbb{R}^6$ has complexity $O^*(r_0^8)$ [14, 28]. A modified version of the preceding analysis then yields the following theorem.

[9]We can make these surfaces independent of π if we add π as a seventh degree of freedom, but then the overall performance of the algorithm deteriorates.

THEOREM 4.4. *The largest-perimeter triangle among those spanned by a set of n points in $\mathbb{R}^3$ can be computed in $O^*(n^{12/5})$ time.*

Largest-height triangle. In this variant, we wish to compute the triangle with largest height among those determined by a set P of n points in $\mathbb{R}^3$, where the height of a triangle is taken to be the largest of its three heights. Here, for each pair p_1, p_2 of points of P, and for a specified height h, we construct the cylinder $C_h(p_1, p_2)$, whose axis passes through p_1 and p_2 and whose radius is h. We run $O(n^2)$ range-fullness queries with these cylinders, and we report that $h^* > h$ if at least one of these cylinders in not full, or $h^* \le h$ otherwise, where h^* is the desired largest height. (Here too one can discriminate between the cases $h^* < h$ and $h^* = h$.)

The efficient implementation of this procedure is carried out as above, except that here the dual representation of these cylinders require only 4 degrees of freedom once h is specified. As in the preceding case, here too the surfaces of P^* also depend on h, so we need a generic parallel procedure for constructing $(1/r_0)$-cuttings for these surfaces, which, however, is not difficult to achieve, since r_0 is a constant. We omit the simple routine details. Since a $(1/r_0)$-cutting in $\mathbb{R}^4$ has complexity $O^*(r_0^4)$ [28], a modified version of the preceding analysis then yields the following theorem.

THEOREM 4.5. *The largest-height triangle among those spanned by a set of n points in $\mathbb{R}^3$ can be computed in $O^*(n^{16/7})$ time.*

Further extensions. We can extend this machinery to higher dimensions, although its performance deteriorates as the dimension grows. The range-fullness problem in $\mathbb{R}^d$, for $d \ge 4$, can be handled in much the same way as in the 3-dimensional case. When extending Claim 4.1, we have an intersection of m convex sets of constant description complexity in $\mathbb{R}^d$, and we can regard the boundary of the intersection as the lower envelope of m $(d-1)$-variate functions of constant description complexity, each representing the boundary of one of the input convex sets in spherical coordinates about some fixed point in the intersection. The complexity of the lower envelope is $O^*(m^{d-1})$ [35]. However, we need to decompose the region below the envelope into elementary cells, and, as already noted, the only known general-purpose technique for doing so is to decompose the entire arrangement of the graphs of the m boundary functions and select the cells below the lower envelope. The complexity of such a decomposition is $O^*(m^{2d-4})$ [14, 28]. This implies that $\zeta(r) = O^*(r^{2d-4})$. The rest of the analysis, including the construction of a good test set, is done in essentially the same manner. Hence, using the machinery of the previous section, we obtain the following theorem.

THEOREM 4.6. *Let P be a set of n points in $\mathbb{R}^d$ for $d \ge 4$, and let Γ be a family of convex ranges of constant description complexity. Then one can construct in near-linear time a data structure of linear size so that, for any range $\gamma \in \Gamma$, it can determine in $O^*(n^{1-1/(2d-4)})$ whether γ is full.*

Finding the largest-area triangle in $\mathbb{R}^d$. Let P be a set of n points in $\mathbb{R}^d$ for $d \ge 4$, and consider the problem of finding the largest-area triangle spanned by P. We apply the same method as in the 3-dimensional case, whose main component is a decision procedure which tests $O(n^2)$ cylinders for fullness. A cylinder (with a line as an axis) in $\mathbb{R}^d$ has $2d-1$ degrees of freedom, so the dual representation of our $O(n^2)$ cylinders is as points in $\mathbb{R}^{2d-1}$. The best known bound on the complexity of a $(1/r)$-cutting in this space is $O^*(r^{2(2d-1)-4}) = O^*(r^{4d-6})$ [14, 28]. Applying this bound and the bound in Theorem 4.6, the overall cost of the decision procedure is

$$O^*\left(n^2(n/r)^{1-1/(2d-4)} + nr^{4d-7}\right).$$

Optimizing the value of r, and applying parametric searching, we get an algorithm for the maximum-area triangle in $\mathbb{R}^d$ with running time

$$O^*\left(n^{1+\frac{(4d-9)(4d-7)}{(4d-6)(2d-4)-1}}\right).$$

We can extend the other problems (largest-perimeter or largest-height triangles) in a similar manner, and we can also obtain algorithms for solving higher-dimensional variants, such as computing the largest-volume tetrahedron or higher-dimensional simplices. We omit the straightforward but tedious analysis and the resulting cumbersome-looking bounds.

5. Ray shooting amid balls in 3-space. Let $\mathcal{B}$ be a set of n balls in 3-space. We show how to preprocess $\mathcal{B}$ in near-linear time into a data structure of linear size so that, given a query ray ρ, the first ball that ρ hits can be computed in $O^*(n^{2/3})$ time, improving the general bound $O^*(n^{3/4})$ mentioned in the introduction. As already noted, we use the parametric-searching technique of Agarwal and Matoušek [2], which reduces the problem to that of efficiently testing whether a query segment $s = qz \subset \rho$ intersects any ball in $\mathcal{B}$, where q is the origin of ρ and z is a parametric point along ρ.

Parametric representation of balls and segments. We move to a parametric 4-dimensional space in which balls in 3-space are represented by points so that a ball with center at (a, b, c) and radius r is mapped to the point (a, b, c, r). A segment e, or for that matter, any closed nonempty set $K \subset \mathbb{R}^3$ of constant description complexity, is mapped to a surface σ_K, which is the locus of all points representing balls that touch K but are openly disjoint from K. By construction, σ_K is the graph of a totally defined continuous trivariate function $r = \sigma_K(a, b, c)$, which is semialgebraic of constant description complexity. Moreover, points below (resp., above) σ_K represent balls which are disjoint from K (resp., intersect K).

Moreover, if we view σ_K, for any such set K, as (the graph of) a trivariate function, then $\sigma_K(q)$ is, by definition, the (Euclidean) distance of q from K. Hence, given a collection $\mathcal{K} = \{K_1, K_2, \ldots, K_m\}$ of m sets, the minimization diagram of the surfaces $\sigma_{K_1}, \ldots, \sigma_{K_m}$ (that is, the projection onto the 3-space $r = 0$ of the lower envelope of these surfaces) is the nearest-neighbor Voronoi diagram of $\mathcal{K}$. We use this property later on in deriving a sharp bound on the resulting function $\zeta(\cdot)$.

Building a test set for segment emptiness. Here we use the general recipe for constructing good test sets. The adaptation of the general recipe to the case at hand is straightforward, except that here our input "points" are in fact n balls in $\mathbb{R}^3$. The initial ranges are defined by segments in 3-space, which have 6 degrees of freedom, so the construction is done in parametric 6-space. The sets K_p are now written as K_B for the balls B in the input collection $\mathcal{B}$. For each cell τ of the resulting decomposed arrangement (of a sample of $O(r \log r)$ regions K_B) we define γ_τ as before, namely, as the union of all ranges corresponding to the segments whose dual points lie in τ. Moreover, if we define by E_τ the union (in $\mathbb{R}^3$) of the segments themselves, then γ_τ is the portion of parametric 4-space above σ_{E_τ}, a property that easily follows from the construction.

We define the desired test set Q to consist of all the "empty" ranges γ_τ (with respect to the sample N), as just defined, and argue that Q indeed satisfies all four properties required from a good test set. Again, compactness, shallowness, and containment follow immediately from the analysis in section 3.4. Concerning efficiency, we claim that the complement of the union of any m ranges in Q can be decomposed into $O^*(m^3)$ elementary cells.

Indeed, complement of the union of m ranges, $\gamma_{\tau_1}, \ldots, \gamma_{\tau_m}$, is the region below the lower envelope of the corresponding surfaces $\sigma_{E_{\tau_1}}, \ldots, \sigma_{E_{\tau_m}}$. To decompose this region, it suffices to produce a decomposition of the 3-dimensional minimization diagram of these surfaces and extend each of the resulting cells into a semiunbounded vertical prism, whose "ceiling" lies on the envelope.

The combinatorial complexity of the minimization diagram of a collection $\Sigma = \{\sigma_{E_{\tau_1}}, \ldots, \sigma_{E_{\tau_m}}\}$ of m trivariate functions of constant description complexity is[10] $O^*(m^3)$ [36]. Moreover, as noted above, the minimization diagram is the Euclidean nearest-neighbor Voronoi diagram of Σ.

We can decompose each cell $V_i = V(E_{\tau_i})$ of the diagram (or, more precisely, the portion of the cell outside the union of the E_{τ_i}'s) using its *star-shapedness* with respect to its "site" E_{τ_i}; that is, for any point $p \in V(E_{\tau_i})$, the segment connecting p to its nearest point on E_{τ_i} is fully contained in $V(E_{\tau_i})$. As is easy to verify, this property holds regardless of the shape, or intersection pattern, of the regions in Σ. We first decompose the 2-dimensional faces bounding V_i into elementary cells, using, e.g., an appropriate variant of 2-dimensional vertical decomposition, and then take each such cell ϕ_0 and extend it to a cell ϕ, which is the union of all segments, each connecting a point in ϕ_0 to its nearest point on E_{τ_i}. The resulting cells, obtained by applying this decomposition to all cells of the diagram, form a decomposition of the portion of the diagram outside the union of the E_{τ_i}'s, into a total of $O^*(m^3)$ elementary cells, as desired. The union of the E_{τ_i}'s themselves, being a subcollection of cells of a 3-dimensional arrangement of m regions of constant description complexity, can also be decomposed into $O^*(m^3)$ cells, using standard results on vertical decomposition in three dimensions [36].

To recap, the construction of Q covers each empty segment e by a fairly complex "canonical" empty region E, which has nonetheless constant description complexity. In parametric 4-space, each such region E is mapped to the portion of space above the corresponding surface σ_E; this is the set of all balls that intersect E. The complement of the union of m such ranges is the portion of 4-space below the lower envelope of the corresponding surfaces σ_{E_i}. Using the connection between this envelope and the Voronoi diagram of the E_i's, we are able to decompose (the diagram and thus) the complement of the union into $\zeta(m) = O^*(m^3)$ elementary cells.

Using Lemma 3.4 and the machinery of section 3, in conjunction with the parametric-searching technique of [2], we thus obtain the following theorem.

THEOREM 5.1. *Ray shooting amid n balls in 3-space can be performed in $O^*(n^{2/3})$ time, using a data structure of $O(n)$ size, which can be constructed in $O^*(n)$ time.*

Remark. In the preceding description, we only considered *empty* ranges. If desired, we can extend the analysis to obtain a data structure which also supports "reporting queries" in which we want to report the first k balls hit by a query ray. We omit the details of this straightforward extension.

6. Range-emptiness searching and reporting in the plane.

Fat triangle reporting and emptiness searching. Let $\alpha > 0$ be a fixed constant, and let P be a set of n points in the plane. We wish to preprocess P, in $O^*(n)$ time, into a data structure of size $O(n)$, which, given an α-fat query triangle Δ (which, as we recall, is a triangle all of whose angles are at least α), can determine in $O^*(1)$ time whether $\Delta \cap P = \emptyset$, or it can report in $O^*(1) + O(k)$ time the points of P in Δ, where $k = |P \cap \Delta|$.

[10] This bound holds regardless of how "badly" the regions in Σ are shaped, or how "wildly" they can intersect one another, as long as each of them has constant description complexity.

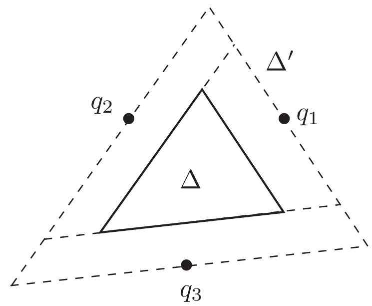

FIG. 2. *The first step in canonizing an empty triangle.*

We note that this problem has a known solution with performance bounds comparable to ours. We will discuss this alternative approach later on in this subsesction.

To obtain our efficient solution we need to construct a good test set Q. We use the following "canonization" process (an ad hoc process, not following the general recipe of section 3.4). As above, we apply the construction to a random sample N of $O(r \log r)$ points of P. For simplicity, we first show how to canonize *empty* triangles, and then we extend the construction to shallow triangles. (As before, the first part suffices for emptiness searching, whereas the second part is needed for reporting queries.) Let Δ be an α-fat empty triangle, which is then also N-empty. We expand Δ homothetically, keeping one vertex fixed and translating the opposite side away until it hits a point q_1 of N. We then expand the new triangle homothetically from a second vertex until the opposite side hits a second point q_2 of N, and then we apply a similar expansion from the third vertex, making the third edge of the triangle touch a third point q_3 of N. We end up with an N-empty triangle Δ', homothetic to, and containing, Δ, each of whose sides passes through one of the points $q_1, q_2, q_3 \in N$. See Figure 2. (It is possible that some of these expansions never hit a point of N, so we may end up with an unbounded wedge or half-plane instead of a triangle. Also, the points q_1, q_2, q_3 need not be distinct.)

Let $\mathcal{D}$ be the set of orientations $\{j\alpha/4 \mid j = 0, 1, \ldots, \lfloor 8\pi/\alpha \rfloor\}$. We turn the side containing q_1 clockwise and counterclockwise about q_1, keeping its endpoints on the lines containing the other two sides until we reach an orientation in $\mathcal{D}$, or until we hit another point of N (which could also be one of the points q_2, q_3), whichever comes first. Each of the new sides forms, with the two lines containing the two other sides, a new (openly) N-empty $(3\alpha/4)$-fat triangle; the union of these two triangles covers Δ'. See Figure 3.

For each of the two new triangles, Δ'', we apply the same construction by rotating the side containing q_2 clockwise and counterclockwise, thereby obtaining two new triangles whose union covers Δ''. We then apply the same construction to each of the four new triangles, this time rotating about q_3. Overall, we get up to eight new triangles whose union covers Δ. Each of these new triangles is $(\alpha/2)$-fat, openly N-empty, and each of its sides either passes through two points of N, or passes through one point of N and has orientation in $\mathcal{D}$. Since $|\mathcal{D}| = O(1/\alpha) = O(1)$, it follows that the overall number of these canonical covering triangles is $O((r \log r)^6) = O^*(r^6)$. (We omit the easy extensions of this step to handle unbounded wedges or half-planes or the cases where the points q_i, or some of the newly encountered points, lie at vertices of the respective triangles.)

We take Q to be the collection of these canonical triangles, and we argue that Q indeed satisfies the properties of a good test set: (i) Compactness: $|Q| = O^*(r^6)$, so

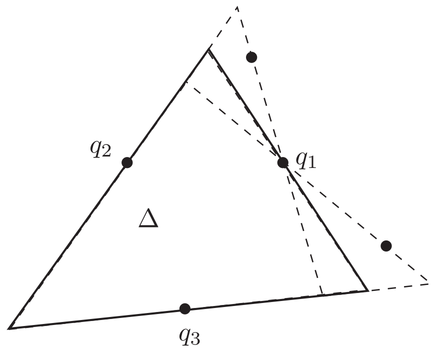

FIG. 3. *The second step in canonizing an empty triangle.*

its size is small. (ii) Shallowness: With high probability, each range in Q is (n/r)-shallow (and, as in section 3.4, we assume that this property does indeed hold). (iii) Containment: By construction, each α-fat empty triangle is contained in the union of at most eight triangles in Q. (iv) Efficiency: Being $(\alpha/2)$-fat, the union of any m triangles in Q has complexity $O(m \log^* m) = O^*(m)$ [9] (see also [23, 32]), so the associated function ζ satisfies $\zeta(m) = O^*(m)$. This, combined with Lemma 3.4 and the machinery of section 3, lead to the following theorem.

THEOREM 6.1. *One can preprocess a set P of n points in the plane in near-linear time into a data structure of linear size so that, for any query α-fat triangle Δ, one can determine, in $O^*(1)$ time, whether $\Delta \cap P = \emptyset$.*

Reporting points in fat triangles. We can extend the technique given above to the problem of reporting the points of P that lie inside any query fat triangle. For this we need to construct a test set that will be good for shallow ranges and not just for empty ones. Using Theorem 2.2, we construct (by random sampling) a shallow $(1/r)$-net $N \subseteq P$ of size $O(r \log r)$. We next canonize every (n/r)-shallow α-fat triangle Δ by the same canonization process used above with respect to the set N. Note that each of the resulting canonical triangles contains (in its interior) the same subset of N as Δ does. By the properties of shallow $(1/r)$-nets, since $|\Delta \cap P| \le n/r$, we have $|\Delta \cap N| = O(\log r)$, so all the resulting canonical triangles are $(c \log r)$-shallow with respect to N for some absolute constant c. Again, since N is a shallow $(1/r)$-net, all the canonical triangles are $(c'n/r)$-shallow with respect to P for another absolute constant c'. Hence, the resulting collection Q of canonical triangles is a good test set for all shallow fat triangles, and we can apply the machinery of section 3 to obtain a data structure of linear size, which can be constructed in near-linear time and which can perform reporting queries in fat triangles in time $O^*(1) + O(k)$, where k is the output size of the query.

An alternative solution for fat triangles. An alternative and well-known solution runs as follows. It suffices to test for emptiness a *semicanonical* fat triangle, as obtained in [32]. This is a triangle with two sides having fixed orientations, chosen from a canonical family $\mathcal{D}$ of $O(1/\alpha)$ orientations and whose third side has an arbitrary orientation. As shown in [32], and depicted in Figure 4, every α-fat triangle Δ is the union of three such semicanonical triangles, so testing Δ for emptiness is equivalent to testing for emptiness each of these covering trianlges

Assume then that we want to test a semicanonical triangle Δ for emptiness. We may assume that the sides e_1, e_2 of Δ with fixed orientations are parallel to the x- and

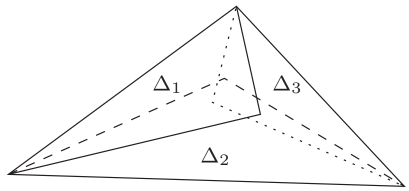

FIG. 4. *Covering a triangle by three semicanonical triangles.*

y-axes, respectively, and that Δ lies above and to the right of these edges. We store the points of P in a 2-dimensional range tree and preprocess each secondary subset for half-plane range emptiness. The storage required by the structure is proportional to the overall size of all secondary subsets, namely, $O(n \log^2 n)$. To test Δ for emptiness, we obtain the subset of P lying above e_1 and to the right of e_2 as the (disjoint) union of $O(\log^2 n)$ secondary sets of the range tree, and we test the half-plane lying below the third edge of Δ for emptiness against each of these subsets in $O(\log n)$ for each test for a total of $O(\log^3 n)$ query time. Thus we obtain a solution with $O(n \log^2 n)$ storage (and $O(n \log^2 n)$ preprocessing) and with $O(\log^3 n)$ query time. Comparing this with our solution, we note that we use less storage but the query time is a little higher.

Range-emptiness searching with semidisks and circular caps. The motivation for studying this problem comes from the following problem, addressed in [16]. We are given a set P of n points in the plane and wish to preprocess it into a data structure of linear size so that, given a query point q and a query line ℓ, one can quickly find the point of P closest to q and lying above ℓ. In the original problem, as formulated in [16], one also assumes that q lies on ℓ, but we will consider, and solve, the more general version of the problem, where q also lies above ℓ.

The standard approach (e.g., as in [3]) yields a solution with linear storage and near-linear preprocessing and query time $O^*(n^{1/2})$. We present a solution with query time $O^*(1)$.

Aronov et al. [8] studied variants of these problems for the case where the points of P are in convex position. They proposed an algorithm that builds a data structure that uses $O(n \log^3 n)$ storage and can efficiently answer queries that seek the nearest point above a line, or the farthest point above a line. The query time is $O(\log n)$. In contrast, we solve the case of nearest-point queries for arbitrary point sets.

Using parametric searching [33], the problem reduces to that of testing whether the intersection of a disk of radius ρ centered at q with the half-plane ℓ^+ above ℓ is P-empty. The resulting range is a circular cap larger than a semidisk (or exactly a semidisk if q lies on ℓ). Again, the main task is to construct a good test set Q for such ranges, which we do by using an ad hoc canonization process, which covers each empty circular cap by $O(1)$ canonical caps, which satisfy the properties of a good test set; in particular, we will have $\zeta(m) = O^*(m)$. (As before, we consider here only the case of emptiness detection, and we will consider the reporting problem later.)

To construct a test set Q we choose a random sample N of $O(r \log r)$ points of P and build a set of canonical empty ranges with respect to N. Let $C = C_{c,\rho,\ell}$ be a given circular cap (larger than a semidisk) with center c, radius ρ, and chord supported by a line ℓ. We first translate ℓ in the direction which enlarges the cap until either its portion within the disk D of the cap touches a point of N or ℓ leaves D. See Figure 5 (left). In the latter case, C is contained in a complete N-empty disk, and it is fairly

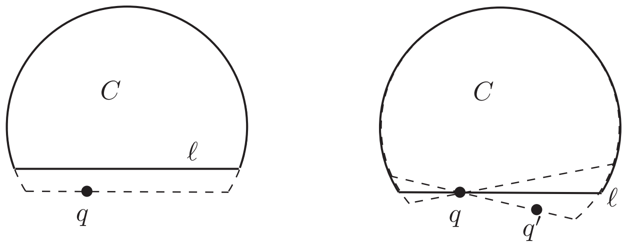

FIG. 5. *The first steps in canonizing an empty cap.*

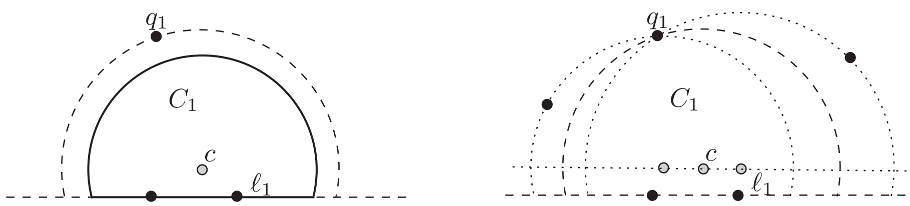

FIG. 6. *The next steps in canonizing an empty cap; the case where* c *lies in* ℓ_1^+.

easy to show that such a disk is contained in the union of at most three canonical N-empty disks, each passing through three points of N or through two diametrically opposite points of N; there are at most $O^*(r)$ such disks, since they are all Deauney disks of N.

Suppose then that the new chord (we continue to denote its line as ℓ) passes through a point q of N, as in the figure. Let $\mathcal{D}$ be a set of $O(1)$ canonical orientations, uniformly spaced and sufficiently dense along the unit circle, for some small constant value α. Rotate ℓ about q in both clockwise and counterclockwise directions until we reach one of the two following events: (i) The orientation of ℓ belongs to $\mathcal{D}$; or (ii) The portion of ℓ within D touches another point of N. In either case, the two new lines, call them ℓ_1, ℓ_2, become canonical—there are only $O^*(r^2)$ such possible lines. Note that our original cap C is contained in the union $C_1 \cup C_2$, where $C_1 = C_{c,\rho,\ell_1}$ and $C_2 = C_{c,\rho,\ell_2}$. Moreover, although the new caps need no longer be larger than a semidisk, they are not much smaller—this is an easy exercise in elementary geometry. See Figure 5 (right).

We next canonize the disk of C (which is also the disk of C_1 and C_2). Fix one of the new caps, say, C_1. Expand C_1 from the center c, keeping the line ℓ_1 fixed until we hit a point q_1 of N (lying in ℓ_1^+). See Figure 6 (left). If c lies in ℓ_1^+, then we move c parallel to ℓ_1 in both directions, again keeping ℓ_1 itself fixed and keeping the circle passing through q_1 until we obtain two circular caps, each passing through q_1 and through a second point of N (if we do not hit a second point, we reach a quadrant, bounded by ℓ and by the line orthogonal to ℓ through q_1). The union of the two new circular caps (each larger than a semidisk) covers C_1. See Figure 6 (right).

If c lies in ℓ_1^-, we move it along the two rays connecting it to the endpoints u_0, v_0 of the chord defined by ℓ_1. As before, each of the motions stops when the circle hits another point of N in ℓ_1^+ or when the motion reaches u_0 or v_0. We claim that C_1 is contained in the union of the two resulting caps. Indeed, let u and v denote the locations of the center at the two stopping placements. We need to show that, for any point $b \in C_1$, we have either $|bu| \le |q_1 u|$ or $|bv| \le |q_1 v|$. If both inequalities did not

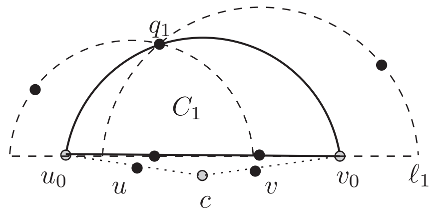

FIG. 7. *The case where c lies in ℓ_1^-; C_1 is contained in the union of the two other caps.*

hold, then both u and v would have to lie on the side of the perpendicular bisector of q_1b containing q_1. This is easily seen to imply that c must also lie on that side, which, however, is impossible (because $|bc| \le |q_1c|$). See Figure 7.

Next, we take one of these latter caps, C', whose bounding circle passes through q_1 and through a second point q_2 of $N \cap \ell_1^+$, and move its center along the bisector of q_1q_2 in both directions, keeping the bounding circle touching q_1 and q_2 and still keeping the line ℓ_1 supporting the chord fixed. We stop when the first of these events takes place: (i) The center reaches ℓ_1, in which case the cap becomes a semidisk (this can happen in only one of the moving directions). (ii) The center reaches the midpoint of q_1q_2. (iii) The bounding circle touches a third point of $N \cap \ell_1^+$. (iv) The central angle of the chord along ℓ_1 is equal to some fixed positive angle $\beta > 0$. The union of the two new caps covers C'. (It is possible that during the motion the moving circle becomes tangent to ℓ_1 and then leaves it, in which case the corresponding final cap is a full disk.)

Similarly, if the center of C' lies on ℓ_1 (which can happen when the motion in the preceding canonization step reaches v_0 or u_0), then we canonize its disk by translating the center to the left and to the right along l_1 until the bounding circle touches another point of $N \cap \ell_1^+$, exactly as in the preceding case (shown in Figure 6 (right)).

Let C'' be one of the four new caps. In all cases C'' is canonical. For the first kind of caps, the stopping condition that defines C'' is (ii) or (iii); then either the circle bounding C'' passes through three points of N or it passes through two diametrically opposite points of N. There are a total of $O^*(r^3)$ such circles, and since C'' is obtained (in a unique manner) by the interaction of one of these circles and one of the $O^*(r^2)$ canonical chord-lines, there is a total of $O^*(r^5)$ such caps. If the stopping condition is (i), the cap is also canonical, because the center of the containing disk is the intersection point of a bisector of two points of N with one of the $O^*(r^2)$ canonical chord-lines, so there is a total of $O^*(r^4)$ such caps. In the case of condition (iv), there are only $O^*(r^2)$ pairs q_1, q_2 and $O^*(r^2)$ canonical chords, for a total of $O^*(r^4)$ caps. Similar reasoning shows that the caps resulting in the second case are also canonical, and their number is $O^*(r^4)$.

We take the test set Q to consist of all the caps of the final forms, and we argue that it satisfies the properties of a good test set: (i) Compactness: $|Q| = O^*(r^5)$, so its size is small. (ii) Shallowness: With high probability, each range in Q is (n/r)-shallow (and we assume that this property does indeed hold). (iii) Containment: Each empty cap is also N-empty, so, by the above canonization process, it is contained in the union of $O(1)$ caps of Q. (iv) Efficiency: Each cap $C \in Q$ is *locally γ-fat* for an appropriate fixed constant $\gamma > 0$ in the terminology of [18], meaning that, for each point $p \in C$ and a disk D centered at p and not fully containing C, we have $area(D \cap C) \ge \gamma \cdot area(D)$. In addition, the boundaries of any two ranges in Q (or of any two circular caps, for that matter) intersect in at most four points, as is easily checked. As follows from the recent analysis of Aronov et al. [9] (see also [18]), the complexity of the union of any m ranges in Q is $O(m2^{O(\log^* m)}) = O^*(m)$. Hence, the complement of the union of any m ranges in Q can be decomposed into $O^*(m)$ elementary cells, making $\zeta(m) = O^*(m)$.

In conclusion, we obtain the following theorem.

THEOREM 6.2. *Let P be a set of n points in the plane. We can preprocess P in near-linear time into a data structure of linear size so that, for any query circular cap C larger than a semidisk, we can test whether $C \cap P$ is empty in $O^*(1)$ time.*

Combining Theorem 6.2 with parametric searching, we obtain the following corollary.

COROLLARY 6.3. *Let P be a set of n points in the plane. We can preprocess P in near-linear time into a data structure of linear size so that, for any query half-plane ℓ^+ and point $q \in \ell^+$, we can find in $O^*(1)$ time the point in $P \cap \ell^+$ nearest to q.*

Remark. The machinery developed in this section also applies to smaller circular caps, as long as they are not too small. Formally, if the central angle of each cap is at least some fixed constant $\alpha > 0$, the same technique holds, so we can test emptiness of such ranges in $O^*(1)$ time, using a data structure which requires $O(n)$ storage and $O^*(n)$ preprocessing. Thus Theorem 6.2 carries over to this scenario, but Corollary 6.3 does not, because we have no control over the "fatness" of the cap, as the disk shrinks or expands when the center of the disk lies in ℓ^-, and once the canonical caps become too thin, the complexity of their union may become quadratic.

Reporting points in semidisks and circular caps. As in the case of fat triangles, we can extend the technique to answer efficiently range reporting queries in semidisks or in sufficiently large circular caps. We use the same canonization process with respect to a random sample N of size $O(r \log r)$ which is a shallow $(1/r)$-net, and we argue, exactly as in the case of fat triangles, that the resulting collection of canonical caps is a good test set for shallow semidisk or larger cap ranges. Applying the machinery of section 3, we obtain a data structure of linear size, which can be constructed in near-linear time, and which can perform reporting queries in semidisks or larger caps in time $O^*(1) + O(k)$, where k is the output size of the query.

7. Approximate range counting.

Given a set P of n points in $\mathbb{R}^d$, a set Γ of semialgebraic ranges of constant description complexity, and a parameter $\delta > 0$, the *approximate range counting* problem is to preprocess P into a data structure such that, for any query range $\gamma \in \Gamma$, we can efficiently compute an approximate count t_γ which satisfies

$$(1-\delta)|P \cap \gamma| \le t_\gamma \le (1+\delta)|P \cap \gamma|.$$

As in most of the rest of the paper, we will only consider the case where the size of the data structure is to be (almost) linear, and the goal is to find solutions with small query times.

The problem has been studied in several recent papers [10, 11, 12, 27] for the special case where P is a set of points in $\mathbb{R}^d$ and Γ is the collection of half-spaces (bounded by hyperplanes). A variety of solutions with near-linear storage were derived; in all of them the dependence of the query cost on n is close to $n^{1-1/\lfloor d/2 \rfloor}$, which, as reviewed earlier, is roughly the same as the cost of half-space range-emptiness queries or the overhead cost of half-space range reporting queries [30].

Remark. Note that in this setup the query range γ remains fixed, and only the count $|P \cap \gamma|$ is to be approximated. An alternative approach, which has recently been studied by Mount and others (see [24]), allows the range γ itself to be approximated, and it computes, more efficiently, the count $|P \cap \gamma'|$ of the points in the approximate range γ'.

The fact that the approximate range counting problem is closely related to range emptiness comes as no surprise, because when $P \cap \gamma = \emptyset$ the approximate count t must be 0, so range emptiness is a special case of approximate range counting. The goal is therefore to derive solutions that are comparable, in their dependence on n, with those that solve emptiness (or reporting) queries. As just noted, this has been accomplished for the case of half-spaces. In this section we extend this technique to the general semialgebraic case.

The simplest solution is to adapt the technique of Aronov and Har-Peled [10], which uses a procedure for answering range-emptiness queries as a "black box." Specifically, suppose we have a data structure $\mathcal{D}(P')$ for any set P' of n' points, which can be constructed in $T(n')$ time, uses $S(n')$ storage, and can determine whether a query range $\gamma \in \Gamma$ is empty in $Q(n')$ time. Using such a black box, Aronov and Har-Peled show how to construct a data structure for n points using $O((\delta^{\lambda-3} + \Sigma_{i=1}^{\lceil 1/\delta \rceil} 1/i^{\lambda-2})S(n)\log n)$ storage and $O((\delta^{\lambda-3} + \Sigma_{i=1}^{\lceil 1/\delta \rceil} 1/i^{\lambda-2})T(n)\log n)$ preprocessing, where $\lambda \geq 1$ is some constant for which $S(n/r) = O(S(n)/r^{\lambda})$ and $T(n/r) = O(T(n)/r^{\lambda})$ for any $r > 1$. Given a range $\gamma \in \Gamma$, the data structure of [10] returns, in $O(\delta^{-2}Q(n)\log n)$ time, an approximate count t_γ, satisfying $(1-\delta)|\gamma \cap P| \leq t_\gamma \leq |\gamma \cap P|$.

The intuition behind this approach is that a range γ, containing m points of P, is expected to contain mr/n points (or, rather, less that one point) in a random sample from P of size r and no points in a sample of size smaller than n/m. The algorithm of [10] then guesses the value of m (up to a factor of $1+\delta$), sets r to be an appropriate multiple of n/m, and draws many (specifically, $O(\delta^{-2}\log n)$ random samples of size r. If γ is empty (resp., nonempty) for many of the samples, then with high probability the guess for m is too large (resp., too small). When we cannot decide either way, we are at the correct value of m (up to a relative error of δ). The actual details of the search are somewhat more contrived; see [10] for those details.

Plugging our emptiness data structures into the machinery of [10], we therefore obtain the following results. In all these applications we can take $\lambda = 1$, so, in the terminology used above, the overall data structure uses $O(\delta^{-2}S(n)\log n)$ storage and $O(\delta^{-2}T(n)\log n)$ preprocessing. (In the bounds stated below, the $O^*(\cdot)$ notation cencerns the dependence of the bounds on n and not on δ.)

Corollary 7.1. *Let P be a set of n points in the plane, and let α, δ be given positive parameters. Then we can preprocess P into a data structure of size $O(\delta^{-2}n\log n)$, in time $O^*(\delta^{-2}n)$, such that, for any α-fat query triangle Δ, we can compute, in $O^*(\delta^{-2})$ time, for any $\delta > 0$, an approximate count t_Δ satisfying $(1-\delta)|\Delta \cap P| \leq t_\Delta \leq |\Delta \cap P|$.*

Corollary 7.2. *Let P be a set of n points in the plane, and let δ be a given positive parameter. Then we can preprocess P into a data structure of size $O(\delta^{-2}n\log n)$, in time $O^*(\delta^{-2}n)$, such that, for any line ℓ, point p on ℓ or above ℓ, and distance d, we can compute, in $O^*(\delta^{-2})$ time, for any $\delta > 0$, an approximate count $t_{\ell,p,d}$ of the exact number $N_{\ell,p,d}$ of the points of P which lie above ℓ and at distance at most d from p so that $(1-\delta)N_{\ell,p,d} \leq t_{\ell,p,d} \leq N_{\ell,p,d}$.*

Corollary 7.3. *Let P be a set of n points in $\mathbb{R}^3$, Γ a collection of convex semialgebraic ranges of constant description complexity, and δ a given positive parameter. Then we can preprocess P into a data structure of size $O(\delta^{-2}n\log n)$, in time*

$O^*(\delta^{-2}n)$, such that, for any query range $\gamma \in \Gamma$, we can compute, in $O^*(\delta^{-2}n^{1/2}\log n)$ time, for any $\delta > 0$, an approximate count t_γ of the number of points of P outside γ, satisfying $(1-\delta)|\gamma^c \cap P| \le t_\Delta \le |\gamma^c \cap P|$.

COROLLARY 7.4. *Let $\mathcal{B}$ be a set of n balls in $\mathbb{R}^3$, and let δ be a given positive parameter. Then we can preprocess $\mathcal{B}$ into a data structure of size $O(\delta^{-2}n\log n)$, in time $O^*(\delta^{-2}n)$, such that, for any query ray ρ, we can compute, in $O^*(\delta^{-2}n^{2/3}\log n)$ time, for any $\delta > 0$, an approximate count t_ρ of the exact number N_ρ of the balls of $\mathcal{B}$ intersected by ρ, satisfying $(1-\delta)N_\rho \le t_\rho \le N_\rho$.*

Remark. Another approach to approximate range counting has been presented in [11, 12] in which, rather than using range-emptiness searching as a black box, one modifies the partition tree of the range-emptiness data structure and augments each of its inner nodes with so-called *relative* (p, ε)*-approximation* sets, which are then used to obtain the approximate count of a range. This approach too can be adapted to yield efficient approximate range counting algorithms for semialgebraic ranges with a slightly improved dependence of their performance on δ. We omit details of such an adaptation in this paper; they can be found in [39].

8. Conclusion. In this paper we have presented a general approach to efficient range-emptiness searching with semialgebraic ranges, and we have applied it to several specific emptiness searching and ray-shooting problems. The present study resolves and overcomes the technical problems encountered in our earlier study [38] and presents more applications of the technique.

Clearly, there are many other applications of the new machinery, and an obvious direction for further research is to "dig them up." In each such problem, the main step would be to design a good test set with associated function $\zeta(\cdot)$ as small as possible, using either the general recipe or an appropriate ad hoc analysis. Many specific instances of this step are likely to generate interesting (and often hard) combinatorial questions. For example, as already mentioned earlier, we still do not know whether the complement of the union of n (congruent) cylinder in $\mathbb{R}^3$ can be decomposed into $O^*(n^2)$ elementary cells.

Appendix A. Proof of Lemma 3.1. The proof is a fairly routine adaptation of the proof in [30]. We employ a variant of the method of Chazelle and Friedman [15] for constructing the cutting. Let Γ' be a random sample of $O(r)$ ranges of Γ, and let E' denote the complement of the union of Γ'. By assumption, E' can be decomposed into at most $O(\zeta(r))$ elementary cells. The resulting collection Ξ of these cells is such that their union clearly contains the complement of the union of Γ. Moreover, the complement of the union of Ξ is the union of the $O(r)$ ranges of Γ'. Hence, Ξ satisfies all three conditions (i)–(iii), but it may fail to be a $(1/r)$-cutting.

This latter property is enforced as in [15] by further decomposing each cell τ of Ξ that is crossed by more than n/r ranges of Γ, using additional subsamples from the surfaces that cross τ. Specifically, for each cell τ of Ξ, let Γ_τ denote the subset of those ranges in Γ that cross τ, and put $\xi_\tau = |\Gamma_\tau| r/n$. If $\xi_\tau > 1$, we sample $q = O(\xi_\tau \log \xi_\tau)$ ranges from Γ_τ, construct the complement of the union of these ranges, decompose it into at most $\zeta(q)$ elementary cells, and clip them to within τ. The resulting collection Ξ' of subcells over all cells τ of the original Ξ clearly satisfies (i). The analysis of [15] (see also [5]) establishes an exponential decay property on the number of cells of Ξ that are crossed by more than $\xi n/r$ ranges as a function of ξ. Specifically, as in [5], the expected number of such cells is $O(2^{-\xi}\mathbb{E}(\zeta(|\Gamma''|))$, where Γ'' is another random sample of Γ, where each member of Γ is chosen with probability $\frac{r}{n\xi}$. This property implies, as usual [15], that Ξ' is (with high probability) a $(1/r)$-cutting, and it also implies that the size of Ξ' is still $O(\zeta(r))$, assuming ζ to be well behaved. Since we have only refined the original cells of Ξ, the number of ranges that cover the complement of the union of the final cells is still $O(r)$. $\square$

REFERENCES

[1] P. K. Agarwal, *Applications of a new space partition technique*, in Proceedings of the 2nd Workshop Algorithms Data Struct., Lecture Notes in Comput. Sci. 519, Springer, Berlin, 1991, pp. 379–392.

[2] P. K. Agarwal and J. Matoušek, *Ray shooting and parametric search*, SIAM J. Comput., 22 (1993), pp. 794–806.

[3] P. K. Agarwal and J. Matoušek, *On range searching with semialgebraic sets*, Discrete Comput. Geom., 11 (1994), pp. 393–418.

[4] P. K. Agarwal and J. Matoušek, *Dynamic half-space range reporting and its applications*, Algorithmica, 13 (1995), pp. 325–345.

[5] P. K. Agarwal, J. Matoušek, and O. Schwarzkopf, *Computing many faces in arrangements of lines and segments*, SIAM J. Comput., 27 (1998), pp. 491–505.

[6] P. K. Agarwal, J. Pach, and M. Sharir, *State of the union (of geometric objects)*, in Proceedings of the Joint Summer Research Conference on Discrete and Computational Geometry: 20 Years Later, Contemp. Math. 452, AMS, 2008, pp. 9–48.

[7] P. K. Agarwal and M. Sharir, *Pipes, cigars, and kreplach: The union of Minkowski sums in three dimensions*, Discrete Comput. Geom., 24 (2000), pp. 645–685.

[8] B. Aronov, P. Bose, E. D. Demaine, J. Gudmundsson, J. Iacono, S. Langerman, and M. Smid, *Data structure for halfplane proximity queries and incremental Voronoi diagrams*, in Proceedings of the 7th Latin American Symposium on Theoretical Information, Lecture Notes in Comput. Sci. 3887, Springer, Berlin, 2006, pp. 80–92.

[9] B. Aronov, M. de Berg, E. Ezra, and M. Sharir, *Improved bounds for the union of locally γ-fat objects*, manuscript, 2010.

[10] B. Aronov and S. Har-Peled, *On approximating the depth and related problems*, SIAM J. Comput., 38 (2008), pp. 899–921.

[11] B. Aronov, S. Har-Peled, and M. Sharir, *On approximate halfspace range counting and relative epsilon-approximations*, in Proceedings of the 23rd ACM Symposium Comput. Geom., 2007, pp. 327–336.

[12] B. Aronov and M. Sharir, *Approximate halfspace range counting*, SIAM J. Comput., 39 (2010), pp. 2704–2725.

[13] S. Basu, R. Pollack, and M.-F. Roy, *Algorithms in Real Algebraic Geometry*, 2nd ed., Springer, New York, 2006.

[14] B. Chazelle, H. Edelsbrunner, L. J. Guibas, and M. Sharir, *A singly exponential stratification scheme for real semialgebraic varieties and its applications*, Theoret. Comput. Sci., 84 (1991), pp. 77–105.

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

[16] O. Daescu, N. Mi, C. Shin, and A. Wolff, *Farthest-point queries with geometric and combinatorial constraints*, Comput. Geom. Theory Appl., 33 (2006), pp. 174–185.

[17] O. Daescu and R. Serfling, *Extremal point queries with lines and line segments and related problems*, Comput. Geom. Theory Appl., 32 (2005), pp. 223–237.

[18] M. de Berg, *Better bounds for the union complexity of locally fat objects*, in Proceedings of the 26th ACM Sympos. on Comput. Geom., 2010, pp. 39–47.

[19] A. Efrat, *The complexity of the union of (α, β)-covered objects*, SIAM J. Comput., 34 (2005), pp. 775–787.

[20] A. Efrat and M. Katz, *On the union of κ-curved objects*, Comput. Geom. Theory Appl., 14 (1999), pp. 241–254.

[21] A. Efrat and M. Sharir, *The complexity of the union of fat objects in the plane*, Discrete Comput. Geom., 23 (2000), pp. 171–189.

[22] E. Ezra, *On the union of cylinders in three dimensions*, Discrete Comput. Geom., 45 (2011), pp. 45–64.

[23] E. Ezra, B. Aronov, and M. Sharir, *Improved bound for the union of fat triangles*, in Proceedings of the 22nd ACM-SIAM Symp. on Discrete Algorithms, 2011, pp. 1778–1785.

[24] G. D. da Fonseca and D. M. Mount, *Approximate range searching: The absolute model*, Comput. Geom., 43 (2010), pp. 434–444.

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

[26] S. Har-Peled and M. Sharir, *Relative p,ε-approximations in geometry*, Discrete Comput. Geom., 45 (2011), pp. 462–496.

[27] H. Kaplan, E. Ramos, and M. Sharir, *Range minima queries with respect to a random permutation, and approximate range counting*, Discrete Comput. Geom., 45 (2011), pp. 3–33.

[28] V. Koltun, *Almost tight upper bounds for vertical decomposition in four dimensions*, J. ACM, 51 (2004), pp. 699–730.

[29] Y. Li, P. M. Long, and A. Srinivasan, *Improved bounds on the sample complexity of learning*, J. Comput. System Sci., 62 (2001), pp. 516–527.

[30] J. Matoušek, *Reporting points in halfspaces*, Comput. Geom. Theory Appl., 2 (1992), pp. 169–186.

[31] J. Matoušek, *Efficient partition trees*, Discrete Comput. Geom., 8 (1992), pp. 315–334.

[32] J. Matoušek, J. Pach, M. Sharir, S. Sifrony, and E. Welzl, *Fat triangles determine linearly many holes*, SIAM J. Comput., 23 (1994), pp. 154–169.

[33] N. Megiddo, *Applying parallel computation algorithms in the design of serial algorithms*, J. ACM, 30 (1983), pp. 852–865.

[34] S. Mohaban and M. Sharir, *Ray shooting amidst spheres in 3 dimensions and related problems*, SIAM J. Comput., 26 (1997), pp. 654–674.

[35] M. Sharir, *Almost tight upper bounds for lower envelopes in higher dimensions*, Discrete Comput. Geom., 12 (1994), pp. 327–345.

[36] M. Sharir and P. K. Agarwal, *Davenport-Schinzel Sequences and Their Geometric Applications,* Cambridge University Press, Cambridge-New York-Melbourne, 1995.

[37] M. Sharir and H. Shaul, *Ray shooting and stone throwing with near-linear storage*, Comput. Geom. Theory Appl., 30 (2005), pp. 239–252.

[38] M. Sharir and H. Shaul, *Ray shooting amid balls, farthest point from a line, and range emptiness searching*, in Proceedings of the 16th ACM-SIAM Symp. on Discrete Algorithms, 2005, pp. 525–534.

[39] H. Shaul, *Range-searching: Emptiness, reporting and approximate counting*, Ph.D. thesis, Tel-Aviv University, Tel-Aviv, Israel, 2011.