Range Counting over Multidimensional Data Streams*

Subhash Suri
Computer Science Dept.
University of California
Santa Barbara, CA 93106
suri@cs.ucsb.edu

Csaba D. Tóth
Computer Science Dept.
University of California
Santa Barbara, CA 93106
toth@cs.ucsb.edu

Yunhong Zhou
Hewlett-Packard Laboratories
1501 Page Mill Road
Palo Alto, CA 94304
yunhong.zhou@hp.com

Abstract

We consider the problem of approximate range counting over streams of d-dimensional points. In the *data stream model*, the algorithm makes a single scan of the data, which is presented in an arbitrary order, and computes a compact summary (called a *sketch*). The sketch, whose size depends on the approximation parameter ε, can be used to count the number of points inside a query range within additive error εn, where n is the size of the stream. We present several results, deterministic and randomized, for both rectangle and halfplane ranges.

Categories and Subject Descriptors: F.2.2 [Discrete Mathematics]: Nonnumerical Algorithms and Problems; G.3.8 [Probability and Statistics]: Probabilistic algorithms; H.2.8 [Database management]: Database Applications

General Terms: Algorithms, Measurement, Theory

Keywords: Data streams, Range counting, ε-approximation

1 Introduction

Data streams have emerged as an important paradigm for processing data that arrives and needs to be processed continuously. For instance, telecom service providers routinely monitor packet flows through their networks to infer usage patterns and signs of attack, or to optimize their routing tables. Financial markets, banks, web servers, and news organizations also generate rapid and continuous data streams.

Many data streams are "geometric" in the sense that each datum is best thought of as a point in some coordinate space. For example, IP packets with four important header fields (source address, destination address, protocol, and time stamp) can be mapped uniquely to points in a 4-dimensional space. The growing adoption of GPS-based devices is making it possible to localize and track users of laptops or PDAs, drivers of cars, trucks, or emergency vehicles. In other applications, such as environmental monitoring, astrophysical or geological measurements, location coordinates are an intrinsic part of the data.

In the data stream model, the goal is to summarize the stream of points seen so far (or recent past) into a compact "sketch," which can answer useful queries approximately [25]. The model is motivated by the reality that in many of the applications mentioned above, data is too large to store in its entirety or to even scan for real-time query processing. For instance, a telecom network manager may wish to understand the *distribution* of network traffic in the source-destination space. He may want to query the packet log: How much TCP data went across a backbone router R that was sent from a subnet S to a subnet D between 1 and 4 PM yesterday? A quick size estimate of the packet log shows that a single Internet Service Provider with one hundred 2.4 Gbps routers will need more than 50 terabytes per day to store just the header information of all the packets. Similarly, after a nightly scan of the sky, an astrophysicist may want to obtain, *in retrospect*, spatial range query estimates of the number of events that occurred in certain regions of interest. Motivated by these applications, we consider in this paper the fundamental problem of approximate range counting over streams of d-dimensional point sets.

1.1 Problem Definition

We assume that the input stream is a sequence of d-dimensional points, $p_1, p_2, \ldots, p_n, \ldots$, where p_1 is the oldest and p_n the most recent point. The total number of points is not known in advance, and the stream can be potentially infinite. We want to maintain a data structure that is able to answer range counting queries in the *real RAM model* on the set of points seen so far. The size of our data structure should ideally be independent of n, the number of points seen so far, or at worst it might be polylogarithmic in n.

Many non-trivial problems, including range counting, are impossible to solve *exactly* in the data stream model. In particular, a classical result by Munro and Paterson [24] shows that any algorithm that computes quantiles of a set of n numbers in at most p passes over the data requires at least $\Omega(n/p)$ space. Therefore, the data stream algorithms aim for ε-approximations using space that is polynomial in $1/\varepsilon$. Indeed, one can compute ε-approximate quantiles in one pass using $O(\frac{1}{\varepsilon}\log(\varepsilon n))$ space [12].

A standard tool for compressing a point set for range counting queries is the ε-approximation introduced by Vap-

*Research by the first two authors was partially supported by NSF grants CCR-9901958 and ANI-9813723.

SCG'04, June 8–11, 2004, Brooklyn, New York, USA.

nik and Chervonenkis [26]. For any range space with finite VC dimension, a random sample of size $O(\frac{1}{\varepsilon^2}\log\frac{1}{\varepsilon\delta})$ is an ε-approximation with probability at least $1-\delta$. An on-line sampling method, such as the reservoir sampling of Vitter [27], can be used to compute a random sample of the data stream almost linearly in time and space with respect to the sample size. In our view, two key challenges for range counting in data streams are: to find deterministic sketching algorithms with small space complexity, and to find randomized sketching algorithms that improve the naïve space complexity $O(\frac{1}{\varepsilon^2}\log\frac{1}{\varepsilon})$.

1.2 Related Work

Exact range counting [1, 19]. Range searching and counting are fundamental problems in computational geometry, with a long history. The main focus has been on achieving fast query time with as little space as possible. However, since the entire input is stored, the space is at least linear, and the data structure construction requires multiple passes over the data. Among the specific results, Chazelle [9] gives an $O(n)$ size data structure that can answer axis-parallel rectangle range counting queries in the plane in $O(\log n)$ time; Matoušek [21] gives an $O(n)$ size data structure that can answer simplex queries in $O(\sqrt{n})$ time. An interested reader should consult one of the surveys [1, 19] for many related results and extensions to d dimensions.

ε-approximations and geometric discrepancy [10, 18]. An *ε-approximation* of a point set P for a range space $\mathcal{Q}$ is a set $S \subset P$ such that, for any range $Q \in \mathcal{Q}$,

$$\left|\frac{|P|}{|S|}\cdot|S\cap Q| - |P\cap Q|\right| \le \varepsilon|P|.$$

A range query is answered by counting how many points of the ε-approximation fall in the query range, and scaling up the answer proportionately.

Matoušek [22] has shown (slightly improving earlier result of Matoušek, Welzl, and Wernisch [23]) that there exists an ε-approximation of size $O(\varepsilon^{-2d/(d+1)})$ for halfspace ranges in $\mathbb{R}^d$, which is the best possible bound apart from constant factors due to discrepancy results of Beck [5] (c.f., [7, 10]). An ε-approximation of size $\tilde{O}(\varepsilon^{-2d/(d+1)})$ can be obtained through $O(\log(\varepsilon n))$ *halving steps* based on the partial coloring of Beck [6] (c.f., [7, 18]), for which no polynomial time algorithm is known.

A random sampling algorithm that picks every point with probability $\Theta(\frac{d}{n\varepsilon^2}\log\frac{1}{\varepsilon})$ provides an ε-approximation with high probability [26]. Chazelle and Matoušek [11, 10] have given a deterministic algorithm to find an $O(\frac{d}{\varepsilon^2}\log\frac{1}{\varepsilon})$ size ε-approximation in $O(d^{3d}\,(\frac{1}{\varepsilon^2}\log\frac{1}{\varepsilon})^d\,n)$ time. Bagchi et al. [3] have recently adapted this algorithm to the data stream model. Their sketching algorithm makes one pass over the input and requires $O\left(\varepsilon^{-2d-2}\cdot\log^{3+\delta} n\cdot\log\frac{1}{\varepsilon}\cdot\log(\frac{1}{\varepsilon}\log n)\right)$ space and time, for some $\delta > 0$.

Range counting over data streams. In one-dimension, Greenwald and Khanna [12] construct a weighted ε-approximation using $O(\frac{1}{\varepsilon}\log(\varepsilon n))$ space. We describe their algorithm in Subsection 2.1, because we use it as a building block for one of our algorithms for axis-aligned box ranges. For a parameter $q \in [0,1]$ given in advance, Manku, Rajagopalan, and Lindsay [16] can find an ε-approximate q-quantile (an element whose rank is in the interval $[(q-\varepsilon)n, (q+\varepsilon)n])$ with a randomized algorithm in $O(\frac{1}{\varepsilon}\log^2\frac{1}{\varepsilon}+\frac{1}{\varepsilon}\log^2\log\frac{1}{\delta}))$ space with probability $1-\delta$.

Finally, Hershberger et al. [15] present a deterministic sketch of size $O(\frac{1}{\varepsilon}\log^{2d-1}R)$ for the *discrete* version of the axis-aligned box range queries: they assume that input points are from the universe $\{1,2,\ldots,R\}^d \subset \mathbb{N}^d$. Two recent survey papers [25, 2] are available on a wide range of data stream problems and models.

1.3 Contribution

We propose several algorithms to compute small size synopses for data streams of d-dimensional points that can give approximate answer to range counting queries.

In Sections 2 and 3 we consider axis-aligned box ranges only. We first present a *deterministic* sketching algorithm, whose space complexity is $O(\frac{1}{\varepsilon^d}\log^d(\varepsilon n))$ in $\mathbb{R}^d$. If we are allowed to make d passes over the data, then we show a deterministic sketch of size $O(\frac{1}{\varepsilon}\log(\varepsilon n)\log^d(\frac{1}{\varepsilon}\log(\varepsilon n)))$. Thus, additional passes can significantly improve the space complexity. The main result in this part of the paper is a *randomized* algorithm that maintains a weighted ε-approximation of size $O(\varepsilon^{\frac{-2d}{d+1}}\log^d(\varepsilon^{\frac{2}{d+1}}n))$ with probability $1-o(1)$.

In Sections 4 and 5, we present randomized algorithms to compute, in one pass, *optimal size* ε-approximations apart from logarithmic factors. For halfspace ranges in $\mathbb{R}^d$, we can maintain an $O(\varepsilon^{\frac{-2d}{d+1}}\log^{d+1}\frac{1}{\varepsilon})$ size weighted ε-approximation; while for axis-aligned box ranges, an $O(\frac{1}{\varepsilon}\log^{2d+1}\frac{1}{\varepsilon})$ size ε-approximation with probability 1/2. The probability can be increased to $1-\delta$, for any $\delta \in (0,1)$, with an additional $\log\frac{1}{\delta}$ factor increase in space.

Even though the latter two sketching algorithms maintain almost optimal size ε-approximations at all times, they both use $O(\frac{1}{\varepsilon}\log\frac{1}{\varepsilon}\log(\varepsilon n))$ *merge steps* which are computationally intensive. The halving step for halfspace ranges, for example, requires $O(\varepsilon^{-2d}\log^{d(d+1)}\frac{1}{\varepsilon})$ work space and time.

1.4 Proof technique

We will use a slight generalization of ε-approximations: A *weighted ε-approximation* of a set P for a range space $\mathcal{Q}$ is a set S where every $s \in S$ also has a weight $w_s \in \mathbb{R}$ such that for every range $Q \in \mathcal{Q}$

$$\left|\sum_{s\in S\cap Q} w_s - |P\cap Q|\right| \le \varepsilon|P|.$$

In our constructions, the sum of weights $\sum_{s\in S} w_s$ always equals the total number of points $|P|$. Specifically, every weighted ε-approximation $S = \{s_i : i \in I\}$ corresponds to a *partition* $\{F_i : i \in I\}$ of the input set P, such that $s_i \in F_i$ and the weight of s_i is $w_i = |F_i|$, $i \in I$.

For a point set $P \subset \mathbb{R}^d$, we call a system $\mathcal{F} = (\{F_i : i \in I\}, \{r_i : i \in I\})$ a *representative system* (RS, for short) if $\{F_i : i \in I\}$ is a partition of P and $r_i \in \mathbb{R}^d$, $i \in I$. Instead of storing the entire RS in memory, we only store the pairs $(|F_i|, r_i)$, $i \in I$, in $O(|I|)$ space.

In our deterministic schemes, the weighted ε-approximation S is simply the set of representatives with the cardinality of the corresponding sets as weights. The error in our estimate is $\left||P \cap Q| - \sum_{i:r_i \in Q} |F_i|\right|$, which equals the number of *misrepresented* points. Therefore, the representatives give a weighted ε-approximation for $\mathcal{Q}$ if the system $\mathcal{F}$ satisfies the following ε-deficiency property for $\mathcal{Q}$:

> A representative system $\mathcal{F}$ over a point set P is ε-**deficient** for $\mathcal{Q}$, if for any range $Q \in \mathcal{Q}$, the total number of points separated from their representative by Q is at most $\varepsilon|P|$. Formally,
>
> $$\sum_{i \in I} |\{p \in F_i : (p \in Q) = (r_i \in Q)\}| \leq \varepsilon|P|.$$

In our randomized schemes, we actually use the representatives only to maintain the partition. Our weighted ε-approximation is a set of samples $S = \{s_i : i \in I\}$, such that each s_i is a point chosen uniformly at random from F_i. Thus, we maintain two RSs on the same partition $\{F_i : i \in I\}$: the system $\mathcal{F}$ and a system of samples $(\{F_i : i \in I\}, \{s_i : i \in I\})$. We show that the representatives form a weighted ε-approximation of $|P \cap Q|$ with high probability if $\mathcal{F}$ is $O(\varepsilon^\alpha)$-deficient for $\mathcal{Q}$ for some $\alpha \in (0, 2)$, and $|F_i| = O(\varepsilon^{2-\alpha-o(1)} n)$ for every $i \in I$.

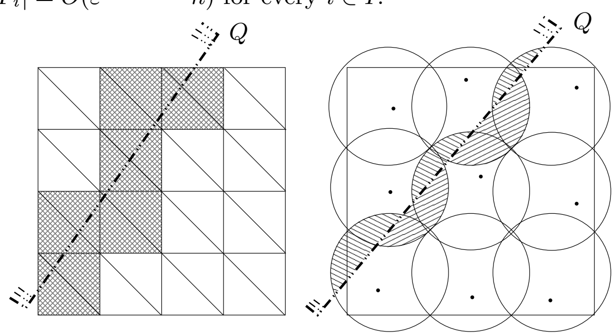

Figure 1. In a simplicial partition, the approximation error comes from the simplices pierced by the boundary of a query Q (left), parts of the sets in a representative system misrepresented for a half-plane Q (right).

The ε-deficiency property for halfspaces in $\mathbb{R}^d$ is also related to *simplicial partitions of low crossing number*, a powerful technique Matoušek [17] developed originally for exact range searching (and, in turn, related to the so called ε-cuttings). Such a simplicial partition of a point set P in $\mathbb{R}^d$ is a system $(\{F_i\}, \{R_i\} : i = 1, 2, \dots, k)$ of size k, $k \in \mathbb{N}$, where

1. $\{F_i : i \in I\}$ is a partition of P such that $|F_i| = \Theta(n/k)$ for every $i \in I$,
2. $F_i \subset R_i$ such that R_i is a simplex, and
3. any hyperplane intersects only $O(k^{\frac{d-1}{d}})$ simplices of $\{R_i : i \in I\}$.

For example, if n points are uniformly distributed in a cube of side length ℓ, then the a triangulation of the integer unit cubes $\{C_1, C_2, \dots, C_{d\ell^d}\}$ give us such a simplicial partition $(\{P \cap C_i\}, \{C_i\} : i = 1, 2, \dots, d\ell^d)$.

If we choose an arbitrary representative point from each set F_i of a Matoušek partition in $\mathbb{R}^d$, then we obtain a representative system satisfying the $\Theta(k^{-1/d})$-deficiency property for halfspaces. These representatives form an ε-approximation for $k = \varepsilon^{-d}$, since the approximation error is the number of points in simplices lying along the boundary of a query halfspace, which totals to $O(k^{\frac{d-1}{d}} \cdot n/k) = O(k^{-1/d} \cdot n)$ [17]. This argument crucially relies on property 3 of such simplicial partitions: If a hyperplane cuts through every simplex R_i, $i \in I$, then these representatives give a 1-approximation. Similarly to our method, random samples chosen from each F_i, $i \in I$, provide a $O(\varepsilon^{\frac{-2d}{d+1}})$ size weighted ε-approximation with high probability for $k = \varepsilon^{\frac{-2d}{d+1}}$ (c.f., jittered sampling [7, 10]). Again: if a hyperplane cuts through all simplices, then a random sample would be only an $O(\sqrt{\varepsilon})$-approximation. Our result (Lemma 3.2) implies that property 3 in the definition of simplicial partitions with low crossing number can be replaced by our $\varepsilon^{\frac{2}{d+1}}$-deficiency property.

2 Deterministic Rectangular Range Counting

2.1 One-dimension: Quantile Summaries

The ε-approximate range counting over one-dimensional data streams is equivalent to the ε-approximate *quantile summaries*. Greenwald and Khanna [12] present a deterministic algorithm for building a data structure of size $O(\frac{1}{\varepsilon}\log(\varepsilon n))$ that can report the rank of any query point $q \in \mathbb{R}$ among the n points seen so far with an error within the interval $[-\varepsilon n, 0]$. We refer to this algorithm as $GK(\varepsilon)$ in the sequel.

They maintain a representative system $\mathcal{F} = (\{F_i\}, \{r_i\} \mid i = 1, 2, \dots, k)$ of size $k = O(\frac{1}{\varepsilon}\log(\varepsilon n))$, but keep only the pairs $(|F_i|, r_i)$, $i = 1, 2, \dots k$, in memory. They also maintain the properties that (i) r_i is the rightmost point of F_i and (ii) $\mathcal{F}$ is ε-deficient for half-lines ranges $\mathcal{Q}^- = \{(-\infty, q] \mid q \in \mathbb{R}\}$. This immediately implies that the $GK(\varepsilon)$ data structure can answer grounded range counting queries $Q \in \mathcal{Q}^-$ with an error within the interval $[-\varepsilon n, 0]$ (there is no over-counting), therefore it can also answer interval range counting queries with an absolute error of at most εn.

We point out two more characteristics of the $GK(\varepsilon)$ algorithm, as we will use it as a building block below. $GK(\varepsilon)$ has two operations: Inserting a new one-element set F_i, and forming the union $F_i \cup F_j$ of two sets F_i and F_j such that $r(F_i \cup F_j) = \max\{r_i, r_j\}$. The parameter ε of $GK(\varepsilon)$ is not necessarily a constant: The algorithm $GK(\varepsilon)$ works within the same complexity bounds as long as εn is monotone increasing.

2.2 Cross Product of Quantile Summaries

In this section we present a deterministic sketching algorithm for axis-parallel box range counting queries. In the plane, it is essentially the cross product of two $GK(\varepsilon)$ sketches. In Section 3, we present more efficient *randomized* sketches based on this algorithm.

THEOREM 2.1. *For a stream of points in the plane, we can deterministically maintain a weighted ε-approximation of size $O(\frac{1}{\varepsilon^2}\log^2(\varepsilon n))$ for axis-aligned rectangle ranges.*

PROOF. We simultaneously run two Greenwald-Khanna algorithms, $GK_x(\frac{\varepsilon}{2})$ and $GK_y(\frac{\varepsilon}{2})$, on the x- and the y-coordinates of the data point. Each generates a partition of the ground set into $O(\frac{1}{\varepsilon}\log(\varepsilon n))$ subsets, $\{G_1, G_2, \dots, G_{k_x}\}$ and $\{H_1, H_2, \dots, H_{k_y}\}$. We store in memory the cardinalities $|G_i|$ and $|H_j|$, and $r(G_i) \in \mathbb{R}$ and $r(H_j) \in \mathbb{R}$, which are the maximal values of each subset according to the x- and y-coordinates respectively.

We also maintain the partition formed by the cross product of $GK_x(\frac{\varepsilon}{2})$ and $GK_y(\frac{\varepsilon}{2})$: $\{G_i \cap H_j \mid i = 1, 2, \ldots, k_x, j = 1, 2, \ldots, k_y\}$. Let the representative of each set $G_i \cap H_j$ be the point $r(G_i \cap H_j) := (r(G_i), r(H_j)) \in \mathbb{R}^2$. This means that all points of $G_i \cap H_j$ lie in a box whose upper right corner is $r(G_i \cap H_j)$ (since $r(G_i)$ is the x-coordinate of the rightmost point of G_i and $r(H_j)$ is the y-coordinate of the highest point of H_j), but the upper right corner of the axis-parallel bounding box of a $G_i \cap H_j$ is not necessarily $r(G_i \cap H_j)$. The two operations of the $GK(\frac{\varepsilon}{2})$ algorithm can be easily represented to update the cross product: (1) when a new point arrives, it is placed in a one-element set G_{k_x} in $GK_x(\frac{\varepsilon}{2})$ and H_{k_y} in $GK_y(\frac{\varepsilon}{2})$, so we insert the one element set $G_{k_x} \cap H_{k_y}$ to the cross product and empty sets $G_{k_x} \cap H_j$ and $G_i \cap H_{k_y}$ for all $j = 1, 2, \ldots, k_y - 1$ and $i = 1, 2, \ldots, k_x - 1$; (2) whenever the sketch $GK_x(\frac{\varepsilon}{2})$ forms the union of two sets G_i and $G_{i'}$, we form the union of $G_i \cap H_j$ and $G_{i'} \cap H_j$ for every $j = 1, 2, \ldots, k_y$.

For a query rectangle $Q = [q_1, q_2] \times [q_3, q_4]$, we report the sum of the cardinalities of sets $G_i \cap H_j$ whose representative $r(G_i \cap H_j)$ lies in Q.

We call a data point p *misrepresented* for a query Q, if $p \in Q$ but the representative of p is not in Q, or vice verse. Due to the $\frac{\varepsilon}{2}$-deficiency property of $GK(\frac{\varepsilon}{2})$, there are at most $\frac{\varepsilon}{2}n$ misrepresented points for any axis-parallel slab query. The number of misrepresented points is at most $2\frac{\varepsilon}{2}n = \varepsilon n$ for any axis-parallel rectangle query Q, because if p is misrepresented for Q then it is also misrepresented for at least one of the axis-parallel slabs spanned by Q. Therefore, the set of representatives $r(G_i \cap H_j)$ with weights $|G_i \cap H_j|$, $\forall i, j$, forms a weighted ε-approximation for axis-parallel rectangles. □

The above theorem generalizes to arbitrary Euclidean space $\mathbb{R}^d$, $d \in \mathbb{N}$. The proof is straightforward and is omitted.

THEOREM 2.2. *For a stream of points in $\mathbb{R}^d$, we can deterministically maintain a weighted ε-approximation of size $O\left(\frac{1}{\varepsilon^d}\log^d(\varepsilon n)\right)$ for axis-parallel box ranges, where the constant of proportionality depends on the dimension d.*

2.3 Range counting in d passes

In many data stream applications, $\Omega(\frac{1}{\varepsilon^d})$ storage can be prohibitive. For instance, with $\varepsilon = 10^{-3}$ or 10^{-5}, the space requirement may be too large even in two dimensions. In this section, we show that by allowing *d passes* over the data, we can significantly improve the space requirement of a deterministic sketching algorithm. Specifically, we describe a data structure that can answer rectangular range counting queries with εn absolute error using $O(\frac{1}{\varepsilon}\text{polylog}(\frac{1}{\varepsilon}, \varepsilon n))$ space. Because our algorithm will make multiple passes, we can assume without loss of generality that the size of the data stream is known to us after the first pass.

THEOREM 2.3. *For n points in the plane, we can deterministically construct in* two passes *over the data a sketch of size*

$$O\left(\frac{1}{\varepsilon}\log(\varepsilon n)\cdot\log^2\left(\frac{1}{\varepsilon}\log(\varepsilon n)\right)\right)$$

that can answer axis-parallel rectangle range counting queries with an absolute error of at most εn.

Proof. Let $\varepsilon_1 \in (0, \varepsilon)$ be a constant to be specified later. In the first pass, we build a sketch according to the x-coordinates of the points with the algorithm $GK(\varepsilon_1)$, and obtain a partition into $\ell = O(\frac{1}{\varepsilon_1}\log(\varepsilon_1 n))$ subsets. Let $r_1, r_2, \ldots, r_\ell$ denote the set of all representative elements, and let $r_0 := -\infty$. By the ε_1-deficiency property, every interval $(r_{i-1}, r_i]$ corresponds to a vertical slab in the plane with at most $\varepsilon_1 n$ data points, $i = 1, 2, \ldots, \ell$.

After the first pass over the data is completed, we build a binary tree T of height $O(\log \ell)$ on the intervals $(r_{i-1}, r_i]$, $i = 1, 2, \ldots, \ell$. Every node of T corresponds to a vertical slab: Every leaf corresponds to the slabs spanned by one interval $(r_{i-1}, r_i]$, every non-leaf node v corresponds to the union of the slabs spanned by the (consecutive) intervals at the descendants of v. Thus, at level i of the tree, we have $\ell/2^i$ slabs, which partition the plane. If we denote by $n_{i,j}$ the number of points in slab j at level i, where $j = 1, 2, \ldots, \ell/2^i$, then we have $\sum_j n_{i,j} = n$, $\forall\, i$. Note that after the first pass, we do not know the exact value of $n_{i,j}$, but $GK(\varepsilon_1)$ gives us an estimate $n^*_{i,j}$ such that $|n^*_{i,j} - n_{i,j}| \le \varepsilon_1 n$.

In the second pass, we build a sketch using the y-coordinates in each vertical slab corresponding to a node of T whenever $n^*_{i,j} > \varepsilon_1 n$, and do nothing otherwise. For a slab containing $n_{i,j}$ points, we apply the algorithm $GK(\frac{\varepsilon_1 n}{n^*_{i,j}})$. This ensures that in each slab, we can answer range counting queries with respect to *horizontal query* slabs with an absolute error of $\varepsilon_1 n$. The size of the secondary data structure for a vertical slab of size $n_{i,j}$ is $O(\frac{n^*_{i,j}}{\varepsilon_1 n}\cdot\log\varepsilon_1 n)$. At level i of the tree T, the total size is

$$\sum_{j=1}^{\ell/2^i} O\left(\frac{n^*_{i,j}}{\varepsilon_1 n}\cdot\log\varepsilon_1 n\right) = O\left(\frac{1}{\varepsilon_1}\log\varepsilon_1 n\right).$$

The total size of all secondary data structures is therefore $O(\log \ell \cdot \frac{1}{\varepsilon_1}\cdot\log(\varepsilon_1 n))$.

It remains to show how we the primary and secondary data structures are for answering a rectangular range counting query $Q = [q_1, q_2] \times [q_3, q_4]$. We locate the intervals $(r_{a-1}, r_a]$ and $(r_b, r_{b+1}]$ containing q_1 and q_2 respectively. We can cover the vertical slab spanned by the interval $(r_a, r_b]$ with $2\log\ell$ disjoint vertical slabs corresponding to nodes of T. In each vertical slab, we query the interval $[q_3, q_4]$, and report the sum of approximate answers.

The error of our answer to a query has two sources: We have ignored the points of Q that lie in the vertical slabs of $(r_{a-1}, r_a]$ and $(r_b, r_{b+1}]$; and we had an absolute error of $\varepsilon_1 n$ in each of the $2\log\ell$ vertical slabs. Our total absolute error is therefore bounded by $(2\log\ell + 2)\cdot\varepsilon_1 n$. Let us choose $\varepsilon_1 = \varepsilon/(2\log\ell + 2)$, and then the absolute error will be at most εn. Thus the total sketch size is bounded by

$$O\left(\log\ell\cdot\frac{1}{\varepsilon_1}\cdot\log(\varepsilon_1 n)\right) = O\left(\log^2\ell\cdot\frac{1}{\varepsilon}\cdot\log(\varepsilon n)\right) = $$
$$= O\left(\frac{1}{\varepsilon}\cdot\log(\varepsilon n)\cdot\log^2\left(\frac{1}{\varepsilon}\log\varepsilon n\right)\right). \quad \square$$

Our construction generalizes to $\mathbb{R}^d$ in a straightforward manner, and we omit the details of the proof.

THEOREM 2.4. *For n points in $\mathbb{R}^d$, we can deterministically construct in d* passes *over the data a sketch of size*

$$O_d\left(\frac{1}{\varepsilon}\log(\varepsilon n)\cdot\log^{2d-2}\left(\frac{1}{\varepsilon}\log(\varepsilon n)\right)\right)$$

that can answer axis-parallel rectangle range counting queries with an absolute error of at most εn.

3 Randomized Range Counting

In our randomized sketching algorithms, we generate a representative system $(\{F_i\}, \{r_i\} : i \in I)$ over the input and for every $i \in I$ we maintain a random sample point $s_i \in F_i$. The points r_i are used to update the representative system as new elements of the data stream arrive, while the random sample points form an ε-approximation to answer range counting queries. We can bound the approximation quality with the following Chernoff bound, which is a slight generalization of the Chernoff bound in [13].

3.1 A Chernoff bound

Our Chernoff bound is independent of the dimension of the space. It is a weighted variant of a classical Chernoff bound in [13].

THEOREM 3.1. *For a natural number m, assume that we are given probabilities $p_1, p_2, \ldots, p_m$, and real-valued weights $w_1, w_2, \ldots, w_m$, where $|w_i| < W$ for some fixed constant $W > 0$. Let $X_1, X_2, \ldots, X_m$ be mutually independent random variables such that $\Pr(X_i = -p_i) = 1 - p_i$ and $\Pr(X_i = 1 - p_i) = p_i$. Set $X = \sum_{i=1}^{m} w_i X_i$ and $M = \sum_{i=1}^{m} |w_i| p_i$. For any $\delta \in (0,1)$, we have*

$$\Pr(|X| \geq \delta M) \leq 2\ e^{-\frac{\delta^2 M}{3W}}.$$

PROOF. Let $M' = \sum_{i=1}^{m} w_i p_i$. For any $t > 0$, we have the following chain of inequalities:

$$\Pr(X \geq \delta M) \;=\; \Pr(e^{tX} \geq e^{t\delta M}) \;\leq\; e^{-t\delta M} E\left[e^{tX}\right]$$
$$= e^{-t\delta M} \prod_{i=1}^{m} E\left[e^{tw_i X_i}\right] \;=\; e^{-t\delta M - tM'} \prod_{i=1}^{m} (1 - p_i + p_i e^{tw_i})$$
$$\leq e^{-t\delta M - tM'} \prod_{i=1}^{m} e^{p_i(e^{tw_i} - 1)} \;=\; e^{-t\delta M - tM' + \sum_{i=1}^{m} p_i (e^{tw_i} - 1)}$$

For $|x| \leq 1$ it is easy to verify the following inequality: $e^x - 1 \leq x + \frac{3}{4}x^2$. Let $t = 2\delta/3W$, then $|tw_i| \leq tW \leq 1$, $\forall\, i$. We bound each exponent of the last expression in the above inequality:

$$\begin{aligned}
& -t\delta M - tM' + \sum_{i=1}^{m} p_i (e^{tw_i} - 1) \\
\leq\;& -t\delta M - tM' + \sum_{i=1}^{m} p_i \left(tw_i + \frac{3}{4} t^2 w_i^2 \right) \\
=\;& -t\delta M - tM' + tM' + \frac{3}{4} \cdot t^2 \sum_{i=1}^{m} p_i w_i^2 \\
\leq\;& -t\delta M + \frac{3}{4} \cdot t^2 M W \;=\; -\frac{\delta^2 M}{3W}.
\end{aligned}$$

Thus $\Pr(X \geq \delta M) \leq e^{-\frac{\delta^2 M}{3W}}$. By replacing X by $-X$, through an almost identical calculation, we have $\Pr(-X \geq \delta M) \leq e^{-\frac{\delta^2 M}{3W}}$. By combining the above two inequalities together, we get the desired result:

$$\Pr(|X| \geq \delta M) = \Pr(X \geq \delta M) + \Pr(-X \geq \delta M) \leq 2 e^{-\frac{\delta^2 M}{3W}}.$$

□

We use this Chernoff bound in the main lemma below, which is the key tool in our analysis.

LEMMA 3.2. *Assume we are given a set P of n points in a range space $(X, \mathcal{Q})$ and an α-deficient representative system $(\{F_i : i \in I\}, \{r_i : i \in I\})$ over P such that $|F_i| \leq \beta n$ for every $i \in I$, where $\alpha, \beta > 0$. For every $i \in I$, let s_i be a sample point chosen uniformly at random from F_i. Then for any range $Q \in \mathcal{Q}$, the sum $\sum_{i \in I, s_i \in Q} |F_i|$ approximates $|P \cap Q|$ with an absolute error of at most εn with probability at least $1 - 2e^{-\varepsilon^2/3\alpha\beta}$.*

PROOF. Fix a range $Q \in \mathcal{Q}$ from the range space. For every $i \in I$, we split F_i into two disjoint subsets A_i and B_i as follows.

$$\begin{aligned}
A_i &= \{p \in F_i \mid (p \in Q \text{ and } r_i \in Q) \text{ or } (p \notin Q \text{ and } r_i \notin Q)\} \\
B_i &= F_i \setminus A_i.
\end{aligned}$$

That is, r_i represents the points in A_i correctly but it misrepresents the points in B_i. Let $a_i = |A_i|$, $b_i = |B_i|$, $w_i = a_i + b_i$. By the α-deficiency property, $\sum_{i=1}^{k} b_i \leq \alpha n$. We set $p_i = \frac{b_i}{w_i}$ and observe that $s_i \in A_i$ with probability $1 - p_i$ and $s_i \in B_i$ with probability p_i.

We associate mutually independent random variables X_i to F_i, $\forall i \in I$, such that $\Pr(X_i = -p_i) = 1 - p_i$ and $\Pr(X_i = 1 - p_i) = p_i$. Notice that the total error of our answer for Q is

$$\begin{aligned}
& |P \cap Q| - \sum_{i \in I, s_i \in Q} |F_i| \\
=\;& \sum_{i : r_i \in Q} \left(\sum_{s_i \notin Q} a_i - \sum_{s_i \in Q} b_i \right) + \sum_{i : r_i \notin Q} \left(\sum_{s_i \notin Q} b_i - \sum_{s_i \in Q} a_i \right) \\
=\;& \sum_{i : r_i \in Q} w_i X_i - \sum_{i : r_i \notin Q} w_i X_i.
\end{aligned}$$

Let $X = \sum_{i=1}^{k} w_i^* X_i$ where $w_i^* = w_i$ if $r_i \in Q$ and $w_i^* = -w_i$ otherwise. Thus $X = |P \cap Q| - \sum_{i \in I, s_i \in Q} |F_i|$ and it denotes the total error of the query answer. Let $M = \sum_{i=1}^{k} |w_i^*| p_i$, then $M = \sum_{i=1}^{k} w_i p_i = \sum_{i=1}^{k} b_i \leq \alpha n$. We set $\delta = \varepsilon n / M$ and $W = \beta n$, and apply Theorem 3.1 (Chernoff bound):

$$\Pr(|X| \geq \varepsilon n) \;\leq\; 2\ e^{-\frac{\delta^2 M}{3W}} \;\leq\; 2\ e^{-\frac{\varepsilon^2 n^2}{3MW}} \;\leq\; 2\ e^{-\frac{\varepsilon^2}{3\alpha\beta}}.$$

□

3.2 Randomized Rectangular Range Counting

We apply Lemma 3.2 to guarantee that a random sample drawn from a cross-product partition of one-dimensional quantile summaries can approximate the number of points in a query rectangle.

THEOREM 3.3. *For a stream of points in $\mathbb{R}^2$, we can maintain a sketch of size*

$$O\left(\varepsilon^{-4/3} \log^2 \left(\varepsilon^{2/3} n\right)\right)$$

that answers axis-parallel rectangle range counting queries with at most εn absolute error with probability $1 - o(1)$.

PROOF. Let $\varphi = \varepsilon^{2/3}$. Similarly to the proof of Theorem 2.1, we maintain the cross-product partition of two data structures $GK_x(\varphi)$ and $GK_y(\varphi)$. By the φ-deficiency property, the size of each set of the partitions G_i and H_j is below φn. We modify the cross-product partition so that if a set $G_i \cap H_j$ would have more than $w = \varphi^2 n / \log^2(\varphi n)$ elements after a union operation, then we keep several copies such that each copy is smaller than w. Since n (and also w) is monotone increasing as new points are streaming in, we merge two small sets later when their union becomes smaller than w.) We obtain a partition $\{F_i \mid i \in I\}$, whose size is still $|I| = O(\varphi^{-2}\log^2(\varphi n))$, asymptotically the same as the size of the cross product partition.

Every F_i is part of a subset $G_j \cap H_h$, and so we let $r_i = r(G_j \cap H_h)$. Since at most φn points are misrepresented by r_i's with respect to any vertical or horizontal slab, at most $2\varphi n$ points are misrepresented by the r_i's with respect to axis-aligned boxes. That is, $(\{(F_i\}, \{r_i\} \mid i = 1, 2, \dots k)$ is (2φ)-deficient for axis-aligned rectangles.

We choose a sample point $s_i \in F_i$ uniformly at random from every F_i. A sample point can easily be maintained under the insertion and union operation. Form every new one-element set F_i, we choose a unique sample $s_i \in F_i$, and for a union $F_h = F_i \cup F_j$, we choose $s_h \in \{s_i, s_j\}$ randomly with probabilities $|F_i|/|F_h|$ and $|F_j|/|F_h|$, respectively. For a query rectangle Q, we report $\sum_{i:s_i \in Q} |F_i|$. Observe that the expected value of our answer is exactly the number of points in Q.

It remains to show that the absolute error is at most εn with high probability. We apply Lemma 3.2 to our partition $\{F_i : i \in I\}$ with $\alpha = 2\varphi$ and $\beta = \varphi^2 \cdot \log^{-2}(\varphi n)$. We conclude, that for every query rectangle Q, the probability that the absolute error $\left|\sum_{i:s_i \in Q} |F_i| - |P \cap Q|\right|$ is above εn is bounded by

$$2e^{-\frac{\varepsilon^2}{6\varphi\beta}} = 2e^{-\frac{\varepsilon^2 \log^2(\varphi n)}{6\varphi^3}} = 2e^{-\frac{\log^2(\varepsilon^{2/3} n)}{6}} = o(1).$$

Thus each rectangle range count query has absolute error bounded by εn with probability $1 - o(1)$. $\square$

The techniques used for $\mathbb{R}^2$ readily generalize to any finite dimensional Euclidean space. The next theorem is an extension of Theorem 3.3.

THEOREM 3.4. *For a stream of points in $\mathbb{R}^d$, we can maintain a sketch of size*

$$O_d\left(\varepsilon^{\frac{-2d}{d+1}} \log^d\left(\varepsilon^{\frac{2}{d+1}} n\right)\right)$$

that answers axis-aligned box range counting queries with εn absolute error with probability $1 - o(1)$.

PROOF. Let $\varphi = \varepsilon^{\frac{2}{d+1}}$ and $w = \varphi^d n / \log^d(\varphi n)$. As in the proof of Theorem 3.3, we maintain the cross-product of d data structures $GK_{x_i}(\varphi)$, one for each coordinate axis. We keep the cardinality of each partition set bounded by w by postponing the union operation if a set size would exceed w. It is easy to verify that the size of the partition is bounded by $O_d((\frac{1}{\varphi}\log(\varphi n))^d) = O_d(\frac{1}{\varphi^d} \cdot \log^d(\varphi n))$. It is also easy to check that the cross-product representative system is $(d\varphi)$-deficient for axis-aligned box ranges. We apply Lemma 3.2 with $\alpha = d\varphi$ and $\beta = \varphi^d \cdot \log^{-d}(\varphi n)$ to the RS and the probability for the absolute error to be over εn is bounded by

$$2e^{-\frac{\varepsilon^2}{3d\cdot\varphi\beta}} = 2e^{-\frac{\varepsilon^2 \log^d(\varphi n)}{3d\cdot\varphi^{d+1}}} = 2e^{-\frac{\log^d(\varepsilon^{2/(d+1)} n)}{3d}} = o(1).$$

$\square$

4 Randomized halfspace Range Counting

Our main result in this section is that we can maintain a weighted ε-approximation of size $O(\varepsilon^{\frac{-2d}{d+1}} \log^{d+1}(\varepsilon^{\frac{2d}{d+1}}))$ with constant probability for half space ranges in d-dimensional Euclidean space. Our algorithm crucially depends on merging two sketches of this size into one, *without significantly worsening the approximation error.*

4.1 The Merge Step

To merge two sketches, we apply a result on matchings with low crossing number (c.f., Thm. 5.17 in [18]), which is tight apart from a constant factor. It is a combination of an argument of Welzl [29] and a packing lemma due to Hausler [14].

LEMMA 4.1. *Given $2k$ points in $\mathbb{R}^d$, there is a matching M of size k such that any hyperplane crosses at most $c_1 \cdot k^{\frac{d-1}{d}}$ edges of M, where $c_1 > 0$ is an absolute constant depending on d only.* $\square$

Such a matching can be constructed by a randomized polynomial-time algorithm, which can be turned into a deterministic algorithm via standard de-randomization [18]. This algorithm can merge two samples of size k into one sample of size k. This deterministic merge step, however, requires $O(k^d)$ memory in d-dimensions.

LEMMA 4.2. *Assume that we are given a representative system $\mathcal{F} = (\{F_i\}, \{r_i\} : i = 1, 2, \dots, 2k\}$, $k \in \mathbb{N}$, over a ground set of size n in $\mathbb{R}^d$. Furthermore, $\mathcal{F}$ is α-deficient for halfspaces and $|F_i| \le \beta n$ for every $i = 1, 2, \dots, 2k$, where $\alpha, \beta > 0$. Then one can construct an $\left(\alpha + c_1 \cdot k^{\frac{d-1}{d}}\beta\right)$-deficient representative system of size k for halfspaces.*

PROOF. We invoke Lemma 4.1 on the set of $2k$ representatives $\{r_i : i = 1, 2, \dots, 2k\}$. We define the partition $\{G_h : h = 1, 2, \dots, k\}$ such that $F_i \cup F_j$ is part of it if and only if $r_i r_j \in M$. For every $G_h = F_i \cup F_j$, $h = 1, 2, \dots, k$, we choose a representative q_h arbitrarily from $\{r_i, r_j\}$. Consider the representative system $\mathcal{G} = (\{G_h\}, \{q_h\} : h = 1, 2, \dots, k)$.

Fix a halfspace $Q \in \mathcal{Q}$, and let Q_0 denote the boundary hyperplane of Q. There are at most αn points misrepresented by r_i's for Q. If a line segment $r_i r_j \in M$ does not cross Q_0, then $q_h \in \{r_i, r_j\}$ misrepresents the same points in $\mathcal{G}$ as r_i and r_j jointly misrepresent in $\mathcal{F}$. If $r_i r_j$ crosses Q_0, however, then $q_h \in \{r_i, r_j\}$ misrepresents at most βn more points in $\mathcal{G}$ than r_i and r_j jointly misrepresent in $\mathcal{F}$. Since at most $c_1 \cdot k^{\frac{d-1}{d}}$ edges of M cross Q_0, the system $\mathcal{G}$ misrepresents at most $\alpha n + c_1 \cdot k^{\frac{d-1}{d}} \beta n$ points for any Q, as required. $\square$

4.2 Sketching for Fixed Size Streams

By a well known result of Vapnik and Chervonenkis [26], a random sample of size $\Theta(\frac{d}{\varepsilon^2}\log\frac{1}{\varepsilon})$ is an ε-approximation for halfspaces in $\mathbb{R}^d$ with high probability. Since an ε_2-approximation of an ε_1-approximation is an $(\varepsilon_1+\varepsilon_2)$-approximation, it is enough to find an ε_1-approximate sketching algorithm whose input is a sequence of $\Theta(\frac{d}{\varepsilon_2^2}\log\frac{1}{\varepsilon_2})$ random samples.

First we show a weak bound where we assume that the total number of points in the stream is known in advance. We will use this algorithm to compress a stream of $\Theta(\frac{d}{\varepsilon_1^2}\log\frac{1}{\varepsilon_1})$ sample points in the next subsection.

THEOREM 4.3. *For a stream of N points in $\mathbb{R}^d$ (where N, $N \geq 2\varepsilon^{\frac{-2d}{d+1}}$, is known in advance), we can construct a sample of size $O(\varepsilon^{\frac{-2d}{d+1}}\log^d(\varepsilon^{\frac{2d}{d+1}}N))$ that can answer half-plane range counting queries such that the absolute error is at most εN with probability at least $1-2\exp(-2\log(\varepsilon^{\frac{2d}{d+1}}N)) \geq 1/2$.*

PROOF. Let $\varphi = \varepsilon^{\frac{2}{d+1}}$ and $k = c_2 \cdot \varphi^{-d}\log^d(\varphi^d N)$, where $c_2 > 0$ is a sufficiently large constant. For convenience, we assume that $N = 2^m k$, with $m \in \mathbb{N}$, (otherwise we append dummy points to the stream to match this constraint). We divide the input stream of N points into blocks of size k, and build a balanced binary tree T on the blocks. T has $m = \log(N/k)$ levels, where level 0 corresponds to the leaves and the root is at level m.

For a node $v \in T$ at level i, let P_v be the set of $2^i \cdot k$ points in descendant blocks of v. For every $v \in T$, we compute inductively a representative system $\mathcal{F}_v$ and a corresponding random sample S_v. For every leaf $v \in T$, we let $\mathcal{F}_v$ to be the trivial partition of P_v into one-element sets and let $S_v = P_v$. At every non-leaf node $v \in T$, we construct $\mathcal{F}_v$ by merging the representative systems of its two children by Lemma 4.2. For every $F_j \in \mathcal{F}_v$, which is generated as $F_j = G_{i_1} \cup G_{i_2}$ from partitions of v's children, we choose the sample point $s(F_j)$ uniformly at random from $\{s(G_{i_1}), s(G_{i_2})\}$.

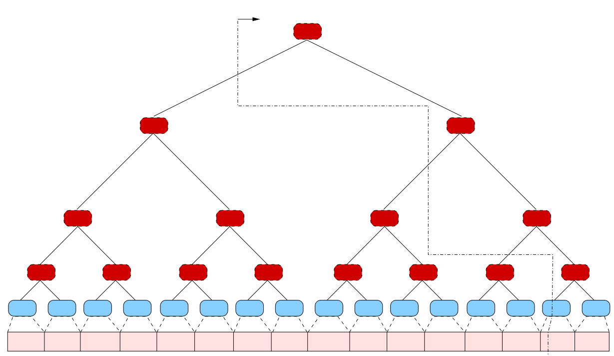

Figure 2. We generate the sketch of a stream of size $N = 2^4 \cdot k$ inductively in 4 levels.

Note that the 2^{m-i} RSs on level i jointly form a RS of the total ground set of size n. The joint RS on the leaf level is 0-deficient. The deficiency increases by $c_1 \cdot k^{-1/d}$ on each level by Lemma 4.2. By induction, the joint RS at level i is $(i \cdot c_1 \cdot k^{-1/d})$-deficient for halfspaces. The root $r \in T$ is at level $m = \log(N/k)$. So the number of points misrepresented for halfspaces by $\mathcal{G}_{root}$ is at most

$$\log\left(\frac{N}{k}\right)\cdot c_1 \cdot k^{-1/d}\cdot N = c_1 \cdot \log\left(\frac{N}{k}\right)\cdot\frac{N}{k^{1/d}}$$
$$= c_1 \cdot \log\left(\frac{\varphi^d N}{c_2 \cdot \log^d(\varphi^d N)}\right)\cdot\frac{N}{c_2^{1/d}\cdot\varphi^{-1}\log(\varphi^d N)}.$$

This is less than φN, if $N \geq 2\varphi^{-d}$ and c_2 is large enough, that is, $\mathcal{G}_{root}$ is φ-deficient for halfspaces.

The size of every set in $\mathcal{G}_{root}$ is $2^m = N/k = \varphi^d N/(c_2 \cdot \log^d(\varphi^d N))$. We apply Lemma 3.2 to $\mathcal{G}_{root}$ with $\alpha = \varphi$ and $\beta = \varphi^d/(c_2\log^d(\varphi^d N))$. By choosing $c_2 \geq 6$, we conclude that the probability that the approximation error $|2^m \cdot |S_v \cap Q| - |P \cap Q||$ is above εN is less than

$$2e^{-\frac{\varepsilon^2 \cdot c_2 \log^d(\varphi^d N)}{3\varphi\cdot\varphi^d}} \leq 2e^{-2\log^d(\varphi^d N)} \leq 2e^{-2\log(\varphi^d N)}$$
$$\leq 2(\varphi^d N)^{-2} \leq 2\cdot 2^{-2} = 1/2. \qquad \square$$

Observe that $O(\varepsilon^{\frac{-2d}{d+1}}\log^d(\varepsilon^{\frac{2d}{d+1}}N))$ is only the size of the final sketch. If we construct the sketch while reading the input, then we need to keep in memory at most one sketch for each of the $\log(N/k)$ different levels of the hierarchy.

COROLLARY 4.4. *For a stream of N points in $\mathbb{R}^d$ (where N, $N \geq 2\varepsilon^{\frac{-2d}{d+1}}$, is known in advance), we can* maintain *at any time n, $1 \leq n \leq N$, a weighted sample of size $O(\varepsilon^{\frac{-2d}{d+1}}\log^d(\varepsilon^{\frac{2d}{d+1}}N)\log(\varepsilon^{\frac{2d}{d+1}}n))$ that can answer half-plane range counting queries such that the absolute error is at most εn with probability at least $1-2\exp(-2\log(\varepsilon^{\frac{2d}{d+1}}N)) \geq 1/2$.*

PROOF. At any time n, $1 \leq n \leq N$, the tree T has at most $\log(n/k) \leq \log(\varphi^d n)$ different levels.

During the inductive construction, we keep in memory at most one representative system and corresponding sample of size k from each level: At the leaf level, we feed points of the data stream into $\mathcal{F}_v$ and S_v. The leaf v is *complete* when all k elements of P_v have arrived. When two children v and w of a non-leaf node u are complete, we construct $\mathcal{F}_u$ and S_u, declare u *complete*, and delete all information regarding v and w. This establishes the claimed space bound. Let S_i denote the level i sample.

The representative systems in memory jointly give an RS of the first n points P_n. We aggregate the samples of different levels into a *weighted sample* such that sample points on level i have weight 2^i. According to Lemma 3.2, the weighted sum $\sum_{i=0}^m 2^i|S_i \cap Q|$ approximates $|P_n \cap Q|$ with an absolute error of at most εn with probability at least $1-2\exp(-2\log(\varphi^d N)) \geq 1/2$. $\square$

4.3 Sketching for General Data Streams

In general, the stream size, n, is not known in advance. A randomized sampling algorithm of Vitter [27] can maintain a sample of size $\Theta(\frac{1}{\varepsilon^2}\log\frac{1}{\varepsilon})$. This algorithm, however, occasionally deletes sample points (it replaces a randomly chosen old sample point by a new one), and so it cannot be fed into an insert-only data stream algorithm. This would be possible in a more general *turnstile* model of data streams [2], where both insertions and deletions are allowed.

To overcome this difficulty, we deploy a simple idea of Manku, Rajagopalan, and Lindsay [16]: After having seen

n_0 points of the stream, we know that the total stream size is at least n_0 and we can choose sample points at a rate $\Omega(\frac{1}{\varepsilon^2}\log\frac{1}{\varepsilon})/n_0$. This results in a non-uniform sample, but we can balance our algorithm so that the output data structure has a uniform error bound.

THEOREM 4.5. *For a stream of points in* $\mathbb{R}^d$, *we maintain a sketch of size* $O(\varepsilon^{\frac{-2d}{d+1}}\log^{d+1}\frac{1}{\varepsilon})$ *that can answer halfspace range counting queries with an absolute error* εn *with probability* $1/2$.

We first describe a sketching algorithm, and then we prove that this algorithm meets the above theoretical bounds.

Algorithm `halfspace-sketch`. Let $K = c_3 \cdot \frac{d}{\varepsilon^2}\log\frac{1}{\varepsilon}$, where c_3 is a sufficiently large constant. Assume, for convenience, that K is a power of 2. We partition the input stream into buckets $B_0, B_1, B_2, \ldots$, etc. The size of B_0 and B_1 is K; while later buckets have exponentially increasing sizes: $|B_i| = 2^{i-1}K$ for $i = 1, 2, \ldots$. From each bucket B_i, we draw a sample D_i of size K. The sampling algorithm of Vitter [28] can choose, if the bucket size $|B_i|$ is known in advance, the sample D_i online in one pass over B_i using only $O(1)$ working space and an expected $O(K)$ time.

Since we cannot afford to keep the entire D_i in memory, we compress it online, while reading the bucket B_i. We feed D_i into the sketching algorithm of Theorem 4.3 and prepare a sketch S_i of size $k = c_2 \cdot \varepsilon^{\frac{-2d}{d+1}}\log^d\frac{1}{\varepsilon} = c_2 \cdot \varphi^{-d}\log^d\frac{1}{\varepsilon}$, where $\varphi = \varepsilon^{\frac{2}{d+1}}$ and c_2 is a large enough constant. The sample set S_i corresponds to a representative system $\mathcal{F}_i$ over D_i. Note that if only an initial portion B_i' of B_i has arrived, then only an initial portion D_i' of D_i is selected. We maintain a partial sketch S_i' of D_i' of size $k\log(K/k)$ according to Corollary 4.4. S_i' corresponds to a representative system $\mathcal{F}_m'$ over D_i'.

A bucket B_i and the sample D_i are *complete* when all points of B_i have arrived and we have chosen all K elements of D_i. At that time, the sample S_i of size k is also complete. Then we prepare a sketch (representative system and sample) of size k for the set $\bigcup_{j=0}^{i} B_j$ for every i inductively: The base case is $\mathcal{F}_0^* = \mathcal{F}_0$ and $S_0^* = S_0$. Then for $i = 1, 2, \ldots$, we construct $\mathcal{F}_i^*$ by merging $\mathcal{F}_{i-1}^*$ and $\mathcal{F}_i$ using Lemma 4.2, and generate the corresponding random sample S_i^* from S_{i-1}^* and S_i.

The n-th point lies in bucket B_m, $m = \lceil\log\lceil\frac{n}{K}\rceil\rceil$. At time n, our current sample set is composed of two parts that jointly form a weighted sample: (1) S_{m-1}^*, each element with weight $2^{m-1}\cdot K/k$. (2) S_m', which consists of at most $\log\frac{K}{k}$ samples of size k coming from different levels of a tree T_m; the weight of every sample point at level j is 2^{m-j-1}. The size of our sketch is therefore $(1+\log(K/k))\cdot k = O\left(\varphi^{-d}\log^{d+1}\frac{1}{\varepsilon}\right) = O\left(\varepsilon^{\frac{-2d}{d+1}}\log^{d+1}\frac{1}{\varepsilon}\right)$.

PROOF (THEOREM 4.5). We show now that Algorithm `halfspace-sketch` maintains a required sketch. At time n, $n \in \mathbb{N}$, the first $m-1 = \lfloor\log\frac{n-1}{K}\rfloor$ buckets are complete, we have seen $n - 2^{m-1}K$ points $B_m' \subset B_m$, and selected samples $D_m' = D_m \cap B_m'$. Let $B = \bigcup_{i=0}^{m-1} B_i \cup B_m'$ be the set of n points seen so far and let $D = \bigcup_{i=0}^{m-1} D_i \cup D_m'$ be a weighted sample set where elements of D_0 have unit weight while the weight of every $a \in D_i$, $i = 1, 2, \ldots, m$, is $w_a = 2^{i-1}$. Our current sketch is $S = S_{m-1}^* \cup S_m'$ corresponding to a representative system $\mathcal{F} = \mathcal{F}_{m-1}^* \cup \mathcal{F}_m'$ over D.

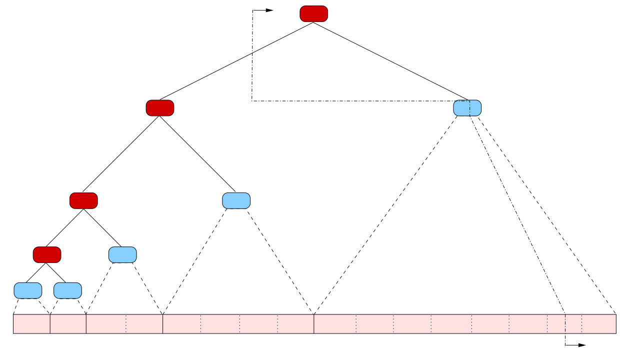

Figure 3. We merge inductively S_i to S_{i-1}^*.

Let $\mu = \log(\varepsilon n/6K))$. The first μ buckets contain the first $\frac{\varepsilon}{3}n$ points of the stream. The remaining points are in $m - \mu = \log\frac{3}{\varepsilon}$ buckets. Our approximation error εn will be composed of three parts: (1) We ignore the first μ buckets, (2) the samples D_i of the last $m-\mu$ buckets jointly form an $\frac{\varepsilon}{3}$-approximation; (3) the sample S is an $\frac{\varepsilon}{3}$-approximation of the weighted set D. The probability that the we err more than εn is, therefore, has two sources: (i) The probability that not all of the last $\log\left(\frac{3}{\varepsilon}\right)$ sets D_i approximate the last $\log\left(\frac{3}{\varepsilon}\right)$ buckets B_i with $\frac{\varepsilon}{3}|B_i|$ error and (ii) the sample S does not approximate D.

Analysis. Every sample D_i is an $\frac{\varepsilon}{6}$-approximation of B_i for halfspaces with probability at least $1 - \frac{\varepsilon}{4}$ for $i = 0, 1, \ldots, m-1$ if c_3 is large enough (c.f., [10], Thm 4.9). D_m' is not necessarily an ε-approximation of B_m', but by analogous argument, $|Q \cap D_m'|\cdot 2^m$ approximates $|Q \cap B_m'|$ with an absolute error of at most $\frac{\varepsilon}{6}|B_m|$ with probability at least $1 - \frac{\varepsilon}{4}$. Therefore, assuming this approximation quality for the last $\log\frac{3}{\varepsilon}$ buckets with probability at least $1 - \frac{\varepsilon}{4}\log\left(\frac{3}{\varepsilon}\right) \geq 3/4$, the weighted set D is a $\frac{2\varepsilon}{3}$-approximation. That is, for any halfspace query Q,

$$\left|\sum_{a\in Q\cap D} w_a - |Q\cap B|\right| \leq \sum_{i=0}^{\mu-1}|B_i| + \sum_{i=\mu}^{m-1}\frac{\varepsilon}{6}|B_i| + \frac{\varepsilon}{6}|B_m| \leq \left(\frac{1}{3}+\frac{1}{6}+\frac{1}{6}\right)\varepsilon n = \frac{2\varepsilon n}{3}.$$

S_i^* is a result of inductive merges of $i+1$ sample sets $S_0, S_1, \ldots, S_i$. Following the proof of Theorem 4.3, each $\mathcal{F}_i$, $i = 0, 1, \ldots, m-1$, is φ-deficient for halfspaces. Similarly, $\mathcal{F}_m'$ is also φ-deficient for halfspaces over the weighted D_m'. This implies that the union of these RSs is φ-deficient over the weighted set D.

The RSs $\mathcal{F}_{i-1}^*$ and $\mathcal{F}_i$ together cover the set $\bigcup_{j=0}^{i} D_i$ of total weight 2^iK, for $i = 1, 2, \ldots, m-1$. By Lemma 4.2, merging these two RSs increases the number of misrepresented points by at most $c_1 \cdot k^{\frac{d-1}{d}}(2^iK/k) = c_1\cdot k^{-1/d}\cdot 2^iK \leq \varphi 2^iK$. The total number of misrepresented points at $\mathcal{F}_{m-1}^*$ is therefore $\varphi\cdot(|\bigcup_{j=0}^{m-1}D_j| + 2\cdot 2^{m-1}K) \leq 3\varphi\cdot|\bigcup_{j=0}^{m-1}D_j|$.

$\mathcal{F} = \mathcal{F}_{m-1}^* \cup \mathcal{F}_m'$ is (3φ)-deficient for halfspaces and the total weight of each set in $\mathcal{F}$ is at most $2^{m-1}\cdot K/k \leq n/k = (\varphi^d\log^{-d}\frac{1}{\varepsilon}/c_2)n$. Let us denote the partition in $\mathcal{F}$ by $\{F_s : s \in S\}$, associating a set $F_s \subset D$ to every sample point $s \in S$ such that $s \in F_s$. We apply Lemma 3.2 to the partition $\{F_s : s \in S\}$ with $\alpha = 3\varphi$ and $\beta = (\varphi^d\log^{-d}\frac{1}{\varepsilon})/c_2$. For every query halfspace Q, the probability that

$$\left| \sum_{s \in S \cap Q} \sum_{b \in F_s} w_b - \sum_{a \in Q \cap D} w_a \right| > \frac{\varepsilon n}{3}$$

is bounded by

$$2e^{-\frac{(\varepsilon/3)^2}{3 \cdot 3\varphi \cdot \varphi^d \log^{-d} \frac{1}{\varepsilon} / c_2}} = 2e^{-\frac{c_2 \log^d \frac{1}{\varepsilon}}{81}} \leq \frac{1}{2}$$

if c_2 is sufficiently large. $\square$

Using standard techniques, we can extend our result and show that our sketch is a weighted ε-approximation for halfspaces with constant probability at any time n, $n \in \mathbb{N}$.

THEOREM 4.6. *For a stream of points in $\mathbb{R}^d$, we maintain a sketch which is a weighted ε-approximation of size $O(\varepsilon^{\frac{-2d}{d+1}} \log^{d+1} \frac{1}{\varepsilon})$ for halfspaces with probability* $1/2$.

PROOF. Instead of testing our weighted sample S against *all* halfspaces, we use a finite collection of regions. Consider an ε-approximation $R \subset P_n$ of size $\ell = c_4 \varepsilon^{-2} \log \frac{1}{\varepsilon}$. The points R span ℓ^d hyperplanes. There exists a collection $\mathcal{Q}_R$ of ℓ^{2d} regions in $\mathbb{R}^d$, each of which is the union or intersection of d halfspaces spanned by R, such that any ε-approximation for $\mathcal{Q}_R$ is a (2ε)-approximation for halfspaces. (Intuitively, any halfspace Q is sandwiched between two regions $Q_1, Q_2 \in \mathcal{Q}_R$ such that $Q_1 \cap P_n \subseteq Q \cap P_n \subseteq Q_2 \cap P_n$.) We know that our weighted sample approximates range counting queries for every region $Q \in \mathcal{Q}_R$ with an absolute error of at most $d\varepsilon n$ with probability at least $1 - \varepsilon^{5d}$, if c_3 and c_2 are chosen appropriately. It follows that it is a weighted $(2d\varepsilon)$-approximation with $1 - \ell 2d\varepsilon^{5d} > 1/2$ probability at any time $n \in \mathbb{N}$. $\square$

5 Rectangle Range Counting Revisited

The core in the sketching algorithm in Section 4 was a merging step that compressed two representative systems for halfspace ranges into one. In the case of axis-aligned rectangular ranges, we can use a similar merging step that compresses ε-approximations directly.

LEMMA 5.1. *Given a set A of $2k$ points in $\mathbb{R}^d$, we can compute a subset $A' \subset A$ of size at most k which is an $(\log^{2d}(2k)/k)$-approximation of A.*

The proof is based on the combinatorial discrepancy of $2k$ points for axis-aligned boxes and on the Beck-Fiala Theorem [8] (c.f., Thm. 4.14 and remarks in [18]). The application of the Beck-Fiala theorem involves solving k linear programs on $2k$ variables. The term $\log^{2d}(2k)/k$ is not the optimal bound: The best known upper bound (for which only existence proofs are known) is $O(\log^{d+\frac{1}{2}}(2k)\sqrt{\log\log 2k}/2k)$ [18]. The best known lower bound, due to Baker [4], is $\Omega(\log^{\frac{d-1}{2}}(2k) \cdot (\log\log(2k)/\log\log\log(2k))^{\frac{1}{2d-1}}/2k)$, $d \geq 3$.

Equipped with a merging algorithm for two ε-approximation of equal size, we can follow the sketching algorithms of the previous section. Instead of a representative system and sample set, we maintain only one sample set A, which is an ε_1-approximation for some $\varepsilon_1 < \varepsilon$.

THEOREM 5.2. *For a stream of N points in $\mathbb{R}^d$ (where $N \in \mathbb{N}$ is known in advance), we can deterministically construct an ε-approximation for axis-aligned rectangles of size $O(\frac{1}{\varepsilon}\log(\varepsilon N)\log^{2d}(\frac{1}{\varepsilon}\log(\varepsilon N)))$.*

PROOF. We follow the main steps in the proof of Theorem 4.3. We divide the input stream into blocks of size

$$k = c_5 \cdot \frac{1}{\varepsilon}\log(\varepsilon N) \cdot \log^{2d}\left(\frac{1}{\varepsilon}\log(\varepsilon N)\right),$$

where $c_5 > 0$ is a sufficiently large constant depending on the dimension d only. We build a balanced binary tree T on the blocks.

For every $v \in T$, we compute inductively a sample A_v of size at most k. For every leaf $v \in T$, we set $A_v = P_v$. At every non-leaf node $v \in T$, we construct A_v by merging the samples of its two children by Lemma 5.1.

The union of all samples on level i of T jointly forms an $(i\log^{2d}(2k)/2k)$-approximation of the total input. We clearly have a 0-approximation at the leaf level, $i = 0$. Then at each level the approximation quality deteriorates by $\log^{2d}(2k)/2k$ according to Lemma 5.1. The root is at level $\log(N/k) \leq \log \varepsilon N$, so the approximation error of the sample A_{root} is

$$\log(\varepsilon N) \cdot \frac{\log^{2d}(2k)}{2k} N \leq \varepsilon N$$

if c_5 is large enough. $\square$

Note that k is the size of the final sketch only. If we construct the sketch while reading the input, then possibly we need to keep in memory one sketch for each of the $\log(n/k)$ different levels of the hierarchy.

COROLLARY 5.3. *For a data stream of N points in $\mathbb{R}^d$ (where $N \in \mathbb{N}$ is known in advance), we can* maintain *at any time n, $1 \leq n \leq N$, a weighted ε-approximation of size $O(\frac{1}{\varepsilon}\log(\varepsilon N)\log^{2d}(\frac{1}{\varepsilon}\log(\varepsilon N))\log(\varepsilon n))$ for axis-aligned rectangles.*

PROOF. We keep in memory at most one sample from each level: At the leaf level, we feed points of the data stream directly into our sample A_v. The leaf v is *complete* when all k elements of P_v have arrived. When two children v and w of a non-leaf node u are complete, then we construct A_u, declare v *complete*, and delete the samples of u's children.

At any time n, $1 \leq n \leq N$, In our sketching algorithm above, we keep at most $\log(n/k) \leq \log(\varepsilon n)$ samples in memory. The size of each sample set is k, which gives the claimed space bound.

These samples are from disjoint subsets of the first n stream points P_n. We aggregate the samples into a *weighted sample* such that sample points on level i have weight 2^i. Each sample set was an ε-approximation of the corresponding base set, so their weighted union is also an ε-approximation. $\square$

If the stream size n is not known in advance, then we follow the structure of Algorithm `halfspace-sketch`: In a first phase, we construct a weighted $\frac{2\varepsilon}{3}$-approximation D by a sampling algorithm of Vitter [28], then in the second phase we compact these samples by applying iteratively the merge step in Lemma 5.1.

THEOREM 5.4. *For a stream of points in $\mathbb{R}^d$, we can maintain a weighted ε-approximation of size $O(\frac{1}{\varepsilon}\log^{2d+1}\frac{1}{\varepsilon})$ for axis-aligned rectangles with probability at least 1/2.*

Algorithm `Box-sketch`. Let $K = c_3 \cdot \frac{d}{\varepsilon^2}\log\frac{1}{\varepsilon}$, where c_3 is a sufficiently large constant. Assume for convenience that K is a power of 2. We partition the input stream into buckets $B_0, B_1, B_2, \ldots$. The size of B_0 is K, and then $|B_i| = 2^{i-1}K$ for $i = 1, 2, \ldots$. From each bucket B_i, we draw a sample D_i of size K, exactly as in the first phase of Algorithm `halfspace-sketch`.

We compress D_i online by the algorithm of Theorem 5.2 and prepare a sample A_i of size $k = c_5 \cdot \frac{6}{\varepsilon}\log^{2d}\frac{6}{\varepsilon}$, where c_5 is a sufficiently large constant depending on d only. On an initial portion D_i' of D_i, we maintain a partial sample A_i' of D_i' of size $k\log(K/k) = O(\frac{1}{\varepsilon}\log^{2d+1}\frac{1}{\varepsilon})$ according to Corollary 5.3.

When bucket B_i is complete, we prepare the sample A_i^* of size k for the set $\bigcup_{j=0}^{i} D_j$ for every i inductively: The base case is $A_0^* = A_0$. Then for $i = 1, 2, \ldots$, we construct A_i^* by merging A_{i-1}^* and A_i using Lemma 5.1.

PROOF (THEOREM 5.4). We show that the Algorithm `Box-sketch` maintains a required sketch. Let $B = \bigcup_{i=0}^{m-1} B_i \cup B_m'$ and $D = \bigcup_{i=0}^{m-1} D_i \cup D_m'$ similarly to the proof of Theorem 4.5. We have seen that D is a weighted $\frac{2\varepsilon}{3}$-approximation of B with probability at least 1/2 if c_3 is large enough.

A_i^* is a result of inductive merges of $i+1$ sample sets $A_0, A_1, \ldots, A_i$. Following the proof of Theorem 5.2, each A_i for $i = 0, 1, \ldots, m-1$, is an $\frac{\varepsilon}{6}$-approximation for D_i.

By Lemma 5.1, merging A_{i-1}^* and A_i increases the approximation error by $(\log^{2d}(2k)/2k)\cdot 2^i K$. The total approximation error of A_{m-1}^* on the weighted set $\bigcup_{i=0}^{m-1} D_i$ is therefore $\frac{\varepsilon n}{6} + (\log^{2d}(2k)/2k)\cdot 2\cdot 2^{m-1}K \le \frac{\varepsilon n}{6} + \frac{n}{k}\log^{2d}(2k) \le \frac{\varepsilon}{3}n$. Together with A_m', which is a weighted $\frac{\varepsilon}{6}$-approximation of D_m', we have a weighted $\frac{\varepsilon}{3}$-approximation of D, and a weighted ε-approximation of B. $\square$

6 Closing Remarks

The are randomized and deterministic algorithms for the halving steps (Lemma 4.2 and 5.1) in our last two sections. The high computational complexity of the currently known algorithms for these subroutines may be prohibitive for data stream applications. It is a long standing open problem to find efficient exact or approximation algorithms for either of them.

References

[1] P.K. Agarwal. Range searching. In Handbook of Discrete and Computational Geometry (J.E. Goodman and J. O'Rourke, eds.), CRC Press, Boca Raton, 1997, pp. 575–598.

[2] B. Babcock, S. Babu, M. Datar, R. Motwani, and J. Widom. Models and issues in data stream systems. In *Proc. Conf. on Principles of Database Systems.*, ACM Press, 2002, pp. 1–16.

[3] A. Bagchi, A. Chaudhary, D. Eppstein, and M.T. Goodrich. Deterministic sampling and range counting in geometric data streams. *These proceedings.*

[4] R. C. Baker. On irregularities of distribution. II. *J. London Math. Soc. (2)* **59** (1999), 50–64.

[5] J. Beck. Irregularities of distribution. I. *Acta Math.* **159** (1987) (1-2), 1–49.

[6] J. Beck. Irregularities of distribution. II. *Proc. London Math. Soc. (3)* **56** (1988), 1–50.

[7] J. Beck and W. Chen. Irregularities of Distributions. Cambridge University Press, 1987.

[8] J. Beck and T. Fiala, Integer-making theorems, *Discrete Applied Math.* **3** (1981), 1–8.

[9] B. Chazelle. A functional approach to data structures and its use in multidimensional searching. *SIAM J. Computing* **17** (3) (1988), 427–462.

[10] B. Chazelle. The Discrepancy Method. Cambridge University Press, 2000.

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

[12] M. Greenwald and S. Khanna. Space-efficient online computation of quantile summaries. *ACM SIGMOD Record* **30** (2) (2001), 58–66.

[13] T. Hagerup and C. Rüb. A guided tour of Chernoff bounds. *Inf. Proc. Letts.* **33** (1990), 305–308.

[14] D. Haussler. Sphere packing numbers for subsets of the Boolean n-cube with bounded VC dimension. *J. Combin. Theory Ser. A* **69** (2) (1995), 217–232.

[15] J. Hershberger, N. Shrivastava, S. Suri, and Cs.D. Tóth. An adaptive sketching technique for multi-dimensional data streams. In preparation.

[16] G.S. Manku, S. Rajagopalan, and B.G. Lindsay. Random sampling techniques for space efficient online computation of order statistics of large datasets. *ACM SIGMOD Record* **28** (2) (1999), 251-262.

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

[18] J. Matoušek. Geometric Discrepancy: An Illustrated Guide. Springer-Verlag, Berlin, 1999.

[19] J. Matoušek. Geometric range searching. *ACM Computing Surveys* **26** (4) (1994), 422–461.

[20] J. Matoušek. Lectures on Discrete Geomtery. Springer-Verlag, Berlin, 2002.

[21] J. Matoušek. Range searching with efficient hierarchical cuttings, *Discrete Comput. Geom.* **10** (1993), 157–182.

[22] J. Matoušek. Tight upper bounds for the discrepancy of half-spaces. *Discrete Com. Geom.* **13** (1995), 593–601.

[23] J. Matoušek, E. Welzl, and L. Wernisch. Discrepancy and approximations for bounded VC-dimension. *Combinatorica* **13** (1993), 455–466.

[24] J. I. Munro and M. S. Paterson. Selection and sorting with limited storage. *Theoretical Computer Science* **12** (1980) 315-323; 1980.

[25] S. Muthukrishnan. Data streams: Algorithms and applications. Preprint, 2003.

[26] V.N. Vapnik and A.Ya. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and its Applications* **16** (1971), 264–280.

[27] J.S. Vitter. Random sampling with a reservoir. *ACM Trans. Math. Software* **11** (1985), 37–57.

[28] J.S. Vitter. Faster methods for random sampling. *Communications of ACM* **27** (1984), 703–718.

[29] E. Welzl. Partition trees for triangle counting and other range searching problems. in *Proc. 4th Sympos. Comput. Geom.*, ACM Press, 1988, pp. 23–33.