Acta Informatica 37, 355–383 (2001)

Acta Informatica

Analysis of range search for random k-d trees

Philippe Chanzy, Luc Devroye, Carlos Zamora-Cura

School of Computer Science, McGill University, Montreal, Canada H3A 2K6
(e-mail: {luc,czamora}@cs.mcgill.ca)

Received: 27 July 1999 / 2 June 2000

Abstract. We analyze the expected time complexity of range searching with k-d trees in all dimensions when the data points are uniformly distributed in the unit hypercube. The partial match results of Flajolet and Puech are reproved using elementary probabilistic methods. In addition, we give asymptotic expected time analysis for orthogonal and convex range search, as well as nearest neighbor search. We disprove a conjecture by Bentley that nearest neighbor search for a given random point in the k-d tree can be done in $O(1)$ expected time.

1 Introduction

The k-d tree, or k-dimensional binary search tree, was proposed by Bentley in 1975. It is a binary tree in which each record contains k keys, right and left pointers to its subtrees, and an index integer between 1 and k that indicates which key in the record is used for splitting. On any path from the root, splitting is performed in a cyclical fashion as explained below. The index of a node at distance ℓ from the root is $1 + \ell \bmod k$. For $k = 1$, we obtain the standard binary search tree. For $k > 1$, consider a node u having index $j \in \{1, \ldots, k\}$. All nodes in its left subtree are such that their j^{th} key is less than the j^{th} key in u, and all nodes in its right subtree are such that their j^{th} key is greater than or equal to the j^{th} key in u. As the indices are $1, 2, \ldots, k, 1, 2, \ldots$ on any path from the root down, we see that each coordinate axis gets cut in turn, in a cyclical fashion. Insertion, deletion and

Research of the authors was sponsored by NSERC grant A3456. The third author received a DGAPA-UNAM Scholarship.

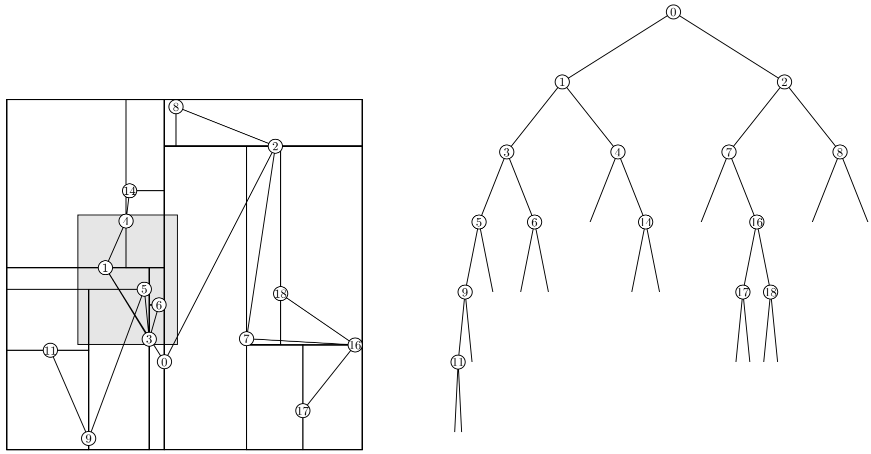

Fig. 1. The k-d tree and its partition of the plane. The query rectangle is shaded

search are implemented as for the standard binary search tree algorithms. These trees are used for a variety of other operations, including orthogonal range searching (report all points within a given rectangle), partial match queries (report all points whose values match a given k-dimensional vector with possibly a number of wild cards, e.g., we may search for all points with values $(a_1, *, *, a_4, a_5, *)$, where $*$ denotes a wild card). Additionally, nearest neighbor searching is greatly facilitated by k-d trees. For orthogonal range searching, a host of particular data structures have been developed, such as the range tree and variations or improvements of it (for surveys, see Bentley and Friedman (1979), Yao (1990), Matoušek (1994), and Agarwal (1997)). However, the k-d tree offers several advantages–it takes $O(kn)$ space for n data points, it is easily updated and maintained, it is simple to implement and comprehend, and it is useful for other operations besides orthogonal range search.

It is instructive to associate with each node the unique rectangle in the partition in which the node falls when it is last added to the tree. Bentley's range search algorithm simply visits recursively all nodes whose rectangle has a nonempty intersection with the query rectangle. In addition, it visits all such rectangles that have no points (and correspond to children of leaves in the tree, see Figure 1—these rectangles will be called leaf regions, and correspond to the nodeless bottom edges in the tree of Figure 1). The query time for orthogonal search time depends upon many factors, such as the location of the query rectangle, and the distribution of the points. One may construct a median k-d tree off-line by splitting each time about the median, thus obtaining a perfectly balanced binary tree, in which search takes $\Theta(\log n)$ worst-case time, and a partial match query takes worst-case time

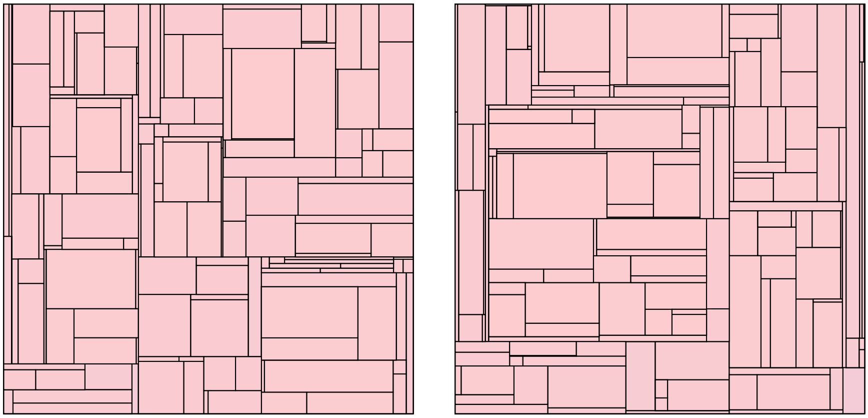

Fig. 2. Two random k-d tree partitions clearly show the elongated rectangles

$O(n^{1-1/k} + N)$, where N is the number of points returned (see, for example, Lee and Wong, 1977). For on-line insertion, balancing is notoriously difficult. If we assume that the data are independent and have a common distribution, then the expected query time is clearly of interest. For standard random binary search trees, it is known (Knuth, 1997; Devroye, 1986; Mahmoud, 1992) that most properties of balanced search trees are inherited: the expected depth of a randomly selected node is about $2 \log n$ and the expected height is $O(\log n)$. One would hope that the random k-d tree, constructed by consecutive insertion of n data points, would also have a performance close to that of the median (off-line) k-d tree. This hope was shattered by Flajolet and Puech (1986). They assumed that the data points are drawn from the uniform distribution on the unit k-cube, and that a partial match query is carried out with s values also drawn uniformly and independently on $[0, 1]$ (so, there are $k - s$ wildcards). The partial match query algorithm is Bentley's range search algorithm with a rectangle having s sides of zero length. On a median k-d tree, partial match query is easily shown to take time about $n^{1-s/k}$. However, Flajolet and Puech stunned the computational geometry community by showing that the expected time performance is $\Theta(n^{1-s/k+\theta(s/k)})$, where $\theta(u)$ is a strictly positive function of $u \in (0, 1)$, with maximum not exceeding 0.07. Thus, random k-d trees behave a bit worse than their balanced counterparts. Surveys of related known probabilistic results are provided by Vitter and Flajolet (1990), and Gonnet and Baeza-Yates (1991).

The purpose of this paper is to complement, extend, and deepen the results of Flajolet and Puech. We will show that the poor behavior of k-d trees for orthogonal range queries is due to the elongated character of most rectangles in the random partitions of the plane defined by the k-d tree. Elongated rectangles have less information (fewer dimensions, if you wish) than squarish rectangles, and will increase search times. We will treat partial match queries as a special case of orthogonal range queries, and derive explicit and tight bounds for random orthogonal range search queries when the query rectangle may have dimensions that depend upon n in an arbitrary fashion. The proofs are entirely probabilistic, rather than analytical, and do not offer explicit constants for expected times but only $\Theta(.)$ results. However, they are short and explain many of the phenomena at work. To illustrate the power of our approach, we finish with results regarding k-d trees queries for membership in random convex sets of a given shape, and look at the expected time for a simple nearest neighbor query. As pointed out in the thesis of Chanzy (1993), nearest neighbor queries based on k-d trees cannot possibly have logarithmic expected times—rather, they grow polynomially in n.

2 Partial match queries

Bentley's algorithm starts the search at the root. At each node, it looks at its index $i \in \{1, \dots, k\}$, and compares the i^{th} key of the current node with the i^{th} range in the search region. If the range is entirely to the left, the search continues only on the left child of the node, if it is entirely to the right, then the search continues only on the right child. Otherwise, the search continues on both children. Given a node u we denote by $\text{left}(u)$ its left child, by $\text{right}(u)$ its right child, and by $\text{index}[u]$ its index.

Let $u_1, u_2, \dots, u_n$, $n \geq 1$ denote the nodes in the k-d tree, and let $U_1, \dots, U_n$ denote the data points, which are i.i.d. and uniformly distributed on $[0,1]^k$. Thus, U_i is the data point corresponding to u_i. The rectangle split by u_i is R_i. Thus, $R_1 = [0,1]^k$. The $n+1$ leaf rectangles (the dangling edges in Figure 1) are also denoted R_i, with the index i now running from $n+1$ to $2n+1$. The collection of rectangles is denoted by $\mathcal{R}_n$. We will denote by T the k-d tree constructed by inserting successively $u_1, u_2, \dots, u_n$ into an initially empty k-d tree. Given a node u in T, we will denote by T_u the subtree of T rooted at u. The dimensions of rectangle R_i are $x_{ij}, 1 \leq j \leq k$. We will consider a query rectangle $Q = [m_1, M_1] \times \cdots \times [m_k, M_k]$ with center $z = (\frac{M_1+m_1}{2}, \dots \frac{M_k+m_k}{2})$. Note that a node u_i is visited by the range search algorithm if and only if the query rectangle Q intersects R_i. A leaf rectangle is visited if its rectangle R_i intersects Q. Let N_n be the time complexity of Bentley's orthogonal range search. Then,

$$N_n = \sum_{i=1}^{2n+1} \mathbb{1}_{[R_i \cap Q \neq \emptyset]} \,.$$

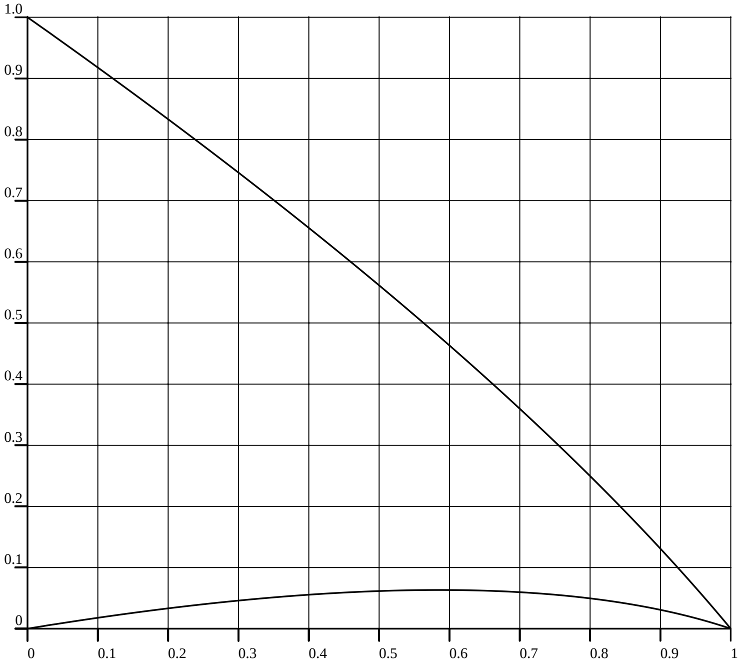

Fig. 3. The top curve is the Flajolet-Puech function $\alpha(.)$. The bottom curve is the function $\theta(.)$

Now, let $1 \leq s \leq k$, and $v_{j_1}, \ldots, v_{j_s} \in [0,1]$. A partial match query asks for all points in $\{u_1, \ldots, u_n\}$ satisfying $u_{ij_t} = v_{j_t}$, for all $1 \leq t \leq s$. We say that the query fixes coordinates $j_1, \ldots, j_s$. We also define L as the set all of points in $[0,1]^k$ whose j_t-th coordinate is equal to v_{j_t}, for all $1 \leq t \leq s$. In a random partial match query, we let the s fixed coordinates be independent and uniformly distributed over $[0,1]$.

The flajolet-puech function. For $0 \leq u \leq 1$, let $\theta(u)$ be the root $\theta \in [0,1]$ of the equation $(\theta+3-u)^u(\theta+2-u)^{1-u} - 2 = 0$. We shall call the function $\alpha(u) = 1-u+\theta(u)$ the Flajolet-Puech function on $[0,1]$. Note in particular that α is decreasing, and that $1-u \leq \alpha(u) \leq 1.07-u$. Equivalently,

$$\alpha(u) = \max_{0 \leq t \leq 1} \left\{ t + 2\left(\frac{1-t}{1-u}\right)^{1-u} \left(\frac{t}{u}\right)^{u} - 2 \right\} .$$

Particular values of interest further on are $\alpha(0) = 1$, $\alpha(1/2) = 0.5616$, $\alpha(1/3) = 0.7162$, $\alpha(2/3) = 0.3949$, $\alpha(1) = 0$.

Theorem 1 (Flajolet and Puech, 1986) *Given is a random k-d tree based on i.i.d. random variables $U_1, \ldots, U_n$, distributed uniformly on $[0,1]^k$. Consider a random partial match query, in which s of the k fields are specified* *with* $k > s \geq 0$. *Let* $N_n^{(s)}$ *be the number of comparisons that Bentley's orthogonal range search performs. Then*

$$\mathbf{E}\left\{N_n^{(s)}\right\} = (c + o(1))n^{\alpha(s/k)} \,,$$

where c *is a constant depending on* s *and* k *only.*

The following proposition is useful in relating random partial match queries to the range search problem.

Proposition 1 *Given is a random k-d tree based on i.i.d. random variables* $U_1, \ldots, U_n$, *distributed uniformly on* $[0,1]^k$. *Consider a random partial match query, in which* $s > 0$ *of the* k *fields are specified. Let* $N_n^{(s)}$ *be the number of comparisons that Bentley's orthogonal range search performs. Let* S *be the set of specified coordinates. Then*

$$\mathbf{E}\left\{N_n^{(s)}\right\} = \mathbf{E}\left\{\sum_{i=1}^{2n+1} \prod_{j \in S} W_{ij}\right\},$$

where $W_{ij}, 1 \leq j \leq k$ *are the widths of the sides of rectangle* R_i *in* $\mathcal{R}_n$.

Proof. Note that $\mathbf{P}\left\{L \cap R_i \neq \emptyset \mid U_1, \ldots, U_n\right\} = \prod_{j \in S} W_{ij}$. Thus we have,

$$\begin{aligned}\mathbf{E}\left\{N_n^{(s)}\right\} &= \mathbf{E}\left\{\sum_{i=1}^{2n+1} \mathbb{1}_{[L \cap R_i \neq \emptyset]}\right\} \\ &= \sum_{i=1}^{2n+1} \mathbf{P}\left\{L \cap R_i \neq \emptyset\right\} = \mathbf{E}\left\{\sum_{i=1}^{2n+1} \prod_{j \in S} W_{ij}\right\}. \ \square\end{aligned}$$

Bibliographical remarks. We know much more about $N_n^{(s)}$ than what is given in Theorem 1. Neininger and Rüschendorf (1999) showed that the first asymptotic term for $\mathbf{Var}\left\{N_n^{(s)}\right\}$ is $\Theta\left(\left(\mathbf{E}\left\{N_n^{(s)}\right\}\right)^2\right)$, and showed that $\left(N_n^{(s)} - \mathbf{E}\left\{N_n^{(s)}\right\}\right) \Big/ \sqrt{\mathbf{Var}\left\{N_n^{(s)}\right\}}$ tends in distribution to a non-degenerate limit law. That is, $N_n^{(s)}$ is asymptotically not concentrated about $\mathbf{E}\left\{N_n^{(s)}\right\}$. Their method of proof uses contractions, and may also be used for analyzing partial match queries for random quadtrees, thus extending results of Flajolet, Gonnet, Puech and Robson (1990, 1992). Partial match queries have also been analyzed for kdt trees, a balanced version of random k-d trees, by Cunto, Lau and Flajolet (1989). For a random k-d tree in which the cut directions are randomly picked, a complete analysis of is given by Duch, Estivill-Castro and Martinez (1998), and Martinez, Panholzer and Prodinger (1998).

3 Orthogonal range search

In this section, we obtain tight upper bounds for the expected complexity for Bentley's range search algorithm.

Lemma 1 *Let $U_1, \ldots, U_n$ be independent uniformly distributed random variables over $[0,1]^k$. Let $\mathcal{R}_n = \{R_1, R_2, \ldots, R_{2n+1}\}$ be the rectangles in the partition defined by the random k-d tree based on $U_1, \ldots, U_n$. Let W_{ij} be the length on the j^{th} coordinate of the i^{th} rectangle. Then,*

$$\mathbf{E}\left\{\sum_{i=1}^{2n+1} W_{i1}\cdots W_{ik}\right\} = 2H_{n+1} - 1,$$

where H_n is the n^{th} harmonic number.

Proof. First, note that for any $1 \le i \le n$, $W_{i1}\cdots W_{ik}$ is the volume $|R_i|$ of the rectangle R_i. Note that if $U_1, \ldots, U_i$ have already been inserted in $[0,1]^k$, and U_{i+1} is a new point, then the combined size of the two rectangles generated by U_{i+1} is equal to the size of the rectangle in the final partition of $[0,1]^k$ in which U_{i+1} falls. Let us denote by $R(U_{i+1})$ this rectangle. Thus, summing over all nodes, we obtain the following identity:

$$\mathbf{E}\left\{\sum_{i=1}^{2n+1} W_{i1}\cdots W_{ik}\right\} = 1 + \sum_{i=0}^{n-1} \mathbf{E}\left\{\mathbf{E}\left\{|R(U_{i+1})| \mid U_1, \ldots, U_i\right\}\right\},$$

where the 1 accounts for the root rectangle. We claim that $\mathbf{E}\{|R(U_{i+1})|\} = \frac{2}{i+2}$. Note that the claim is obviously true for $i = 0$. Now, suppose that $U_1, \ldots, U_i$ have already been inserted in the k-d tree, so that there are $i+1$ external nodes. These external nodes represent the $i+1$ rectangles partitioning $[0,1]^k$. Let these rectangles be $S_1, \ldots, S_{i+1}$, numbered so that the leaves are taken from left to right, in order of appearance as leaves in the k-d tree of $U_1, \ldots, U_i$. Then,

$$\begin{aligned}
\mathbf{E}\{|R(U_{i+1})|\} &= \mathbf{E}\left\{\mathbf{E}\left\{\sum_{\ell=1}^{i+1} \mathbb{1}_{[U_{i+1}\in S_\ell]}\,|S_\ell| \;\Big|\; U_1, \ldots, U_i\right\}\right\} \\
&= \mathbf{E}\left\{\sum_{\ell=1}^{i+1} |S_\ell|\,\mathbf{P}\left\{U_{i+1} \in S_\ell \;\big|\; U_1, \ldots, U_i\right\}\right\} \\
&= \mathbf{E}\left\{\sum_{\ell=1}^{i+1} |S_\ell|^2\right\}.
\end{aligned}$$

Observe that $(|S_1|, \ldots, |S_{i+1}|)$ are jointly distributed as uniform spacings, that is the lengths of the intervals on $[0,1]$ defined by an i.i.d. uniform

$[0,1]$ sample of size i. This is best seen inductively, as the next point added "chooses" a rectangle S_j with probability $|S_j|$, and replaces it by two rectangles, of sizes $U|S_j|$ and $(1-U)|S_j|$ respectively, so that the new rectangle sizes jointly are once again distributed as uniform spacings. All these spacings are identically distributed following a Beta$(1,i)$ distribution. If B is a Beta$(1,i)$ random variable, then we have $\mathbf{E}\{B\} = 1/(i+1)$ and $\mathbf{E}\left\{B^2\right\} = 2/((i+1)(i+2))$. Therefore,

$$\mathbf{E}\{|R(U_{i+1})|\} = (i+1)\,\mathbf{E}\left\{B^2\right\} = \frac{2}{i+2}\,.$$

and thus,

$$1 + \sum_{i=0}^{n-1} \mathbf{E}\{|R(U_{i+1})|\} = 1 + 2(H_{n+1} - 1).\ \square$$

We will also need the following proposition.

Proposition 2 *Given is a random k-d tree based on i.i.d. random variables $U_1,\dots,U_n$, distributed uniformly on $[0,1]^k$. Let W_{ij} be the length of the j^{th} side of rectangle $R_i \in \mathcal{R}_n$. Then, there is a constant $C > 0$, depending on k only, such that*

$$\mathbf{E}\left\{\sum_{i=1}^{2n+1} \mathbb{1}_{\left[\max_{j\in\{1,\dots,k\}} W_{ij} \geq \frac{1}{2}\right]}\right\} \leq C.$$

In other words, the expected number of rectangles with any side greater than $1/2$ does not exceed C.

Proof. For $\ell \geq 1$, let $X^{(\ell)}$ be the product of $\lfloor \ell/k \rfloor$ independent uniform$[0,1]$ random variables. Then, letting ℓ denote the level in the tree,

$$\begin{aligned}
\mathbf{E}\left\{\sum_{i=1}^{2n+1} \mathbb{1}_{\left[\max_{j\in\{1,\dots,k\}} W_{ij} \geq \frac{1}{2}\right]}\right\} &\leq \mathbf{E}\left\{\sum_{i=1}^{2n+1}\sum_{j=1}^{k} \mathbb{1}_{\left[W_{ij} \geq \frac{1}{2}\right]}\right\} \\
&\leq \sum_{\ell=1}^{\infty} 2^\ell k\, \mathbf{E}\left\{\mathbb{1}_{\left[X^{(\ell)} \geq \frac{1}{2}\right]}\right\} \\
&\leq \sum_{\ell=1}^{\infty} 2^\ell k\, \mathbf{E}\left\{\left(X^{(\ell)}\right)^p\right\} 2^p \\
&\leq k 2^p (p+1) \sum_{\ell=1}^{\infty} \left(\frac{2}{(p+1)^{\frac{1}{k}}}\right)^\ell,
\end{aligned}$$

for any $p \geq 0$. The last expression is finite, for example, if we take $p = 2^k$, as $k \geq 2$. $\square$

Theorem 2 *Given is a random k-d tree based on i.i.d. random variables $U_1, \ldots, U_n$, distributed uniformly on $[0,1]^k$. Let Q be a random query rectangle of dimensions $\Delta_1 \times \cdots \times \Delta_k$ (which are deterministic functions of n taking values in $[0,1]$), with center at Z which is uniformly distributed on $[0,1]^k$, and independent of $U_1, \ldots, U_n$. Let N_n be the number of comparisons that Bentley's orthogonal range search algorithm performs. Then, there exist constants $\gamma', \gamma > 0$ depending upon k only such that*

$$\gamma' \le \frac{\mathbf{E}\{N_n\}}{\left(\log n + \sum_{\substack{S \subseteq \{1,\ldots,k\} \\ |S|<k}} \left(\prod_{j \notin S} \Delta_j\right) \, n^{\alpha(|S|/k)}\right)} \le \gamma.$$

Proof. Note that given $U_1, \ldots, U_n$, the probability that Q intersects R_i is the probability that Z has some coordinate Z_j that is within distance $\Delta_j/2$ of R_i, and this probability is clearly bounded by the volume of R_i expanded by Δ_j in the j-th direction, for all j. Thus,

$$\begin{aligned}
\mathbf{E}\{N_n\} &\le \mathbf{E}\left\{\sum_{i=1}^{2n+1} \prod_{j=1}^{k} (W_{ij} + \Delta_j)\right\} \\
&= \sum_{S \subseteq \{1,\ldots,k\}} \left(\prod_{j \notin S} \Delta_j\right) \mathbf{E}\left\{\sum_{i=1}^{2n+1} \prod_{j \in S} W_{ij}\right\} \\
&\le C \sum_{S \subseteq \{1,\ldots,k\} : |S|<k} \left(\prod_{j \notin S} \Delta_j\right) n^{\alpha(|S|/k)} + 2H_{n+1} - 1
\end{aligned}$$

for some constant $C > 0$ and for all n large enough, by Theorem 1 and Lemma 1. For the lower bound notice that,

$$\begin{aligned}
\mathbf{E}\{N_n\} &\ge \mathbf{E}\left\{\sum_{i=1}^{2n+1} \mathbb{1}_{[Q \cap R_i \neq \emptyset]} \mathbb{1}_{[\forall j \in \{1,\ldots,k\} : W_{ij} < 1/2]}\right\} \\
&\ge \mathbf{E}\left\{\sum_{i=1}^{2n+1} \prod_{j=1}^{k} \left(W_{ij} + \frac{\Delta_j}{2}\right) \mathbb{1}_{[\forall \ell \in \{1,\ldots,k\} : W_{i\ell} < 1/2]}\right\} \\
&= \mathbf{E}\left\{\sum_{i=1}^{2n+1} \prod_{j=1}^{k} \left(W_{ij} + \frac{\Delta_j}{2}\right)\right\}
\end{aligned}$$

$$
\begin{aligned}
&- \mathbf{E}\left\{\sum_{i=1}^{2n+1}\prod_{j=1}^{k}\left(W_{ij}+\frac{\Delta_j}{2}\right)\mathbb{1}_{[\exists \ell\in\{1,\dots,k\}:W_{i\ell}\geq 1/2]}\right\}\\
&= \sum_{S\subseteq\{1,\dots,k\}}\prod_{j\notin S}\frac{\Delta_j}{2}\,\mathbf{E}\left\{\sum_{i=1}^{2n+1}\prod_{j\in S}W_{ij}\right\}\\
&\quad - \sum_{S\subseteq\{1,\dots,k\}}\prod_{j\notin S}\frac{\Delta_j}{2}\,\mathbf{E}\left\{\sum_{i=1}^{2n+1}\prod_{j\in S}W_{ij}\,\mathbb{1}_{[\exists \ell\in\{1,\dots,k\}:W_{i\ell}>1/2]}\right\}.
\end{aligned}
$$

We can bound the second term above for any given $S \subseteq \{1,\dots,k\}$ as follows:

$$
\begin{aligned}
&\mathbf{E}\left\{\sum_{i=1}^{2n+1}\prod_{j\in S}W_{ij}\,\mathbb{1}_{[\exists \ell\in\{1,\dots,k\}:W_{i\ell}>1/2]}\right\}\\
&\qquad\leq \mathbf{E}\left\{\sum_{i=1}^{2n+1}\mathbb{1}_{\left[\max_{j\in\{1,\dots,k\}}W_{ij}\geq\frac{1}{2}\right]}\right\}\leq C',
\end{aligned}
$$

by the previous proposition. The result follows by Theorem 1 and Lemma 1. □

Put differently, there exists a constant $\gamma > 0$ depending upon k only such that

$$
\mathbf{E}\{N_n\}\leq\gamma\left(n|Q|+\sum_{s=1}^{k-1}n^{\alpha(s/k)}\sum_{\substack{S\subseteq\{1,\dots,k\}\\|S|=s}}\left(\prod_{j\notin S}\Delta_j\right)+\log n\right),
$$

where the first term accounts for the number of points returned in the query (which is unavoidable), the last term represents the height of the tree (and is unavoidable as well), and the other terms represent contributions from lower-dimensional searches. By Theorem 1, each of these is necessary as well. Note in particular that for $s\in\{0,1,\dots,k\}$, there exists a choice for the Δ_j's that makes a different term in the upper bound dominate. That is, for any given $s<k$, there is a selection of Δ_j's such that the upper bound is $O\left(n^{\alpha(s/k)}\right)$. Just set $\Delta_j=0$, $j\leq s$, $\Delta_j=1$, $j>s$. For $s=k$, the complexity is $O(\log n)$ when all values of Δ_j are zero.

Two-dimensional special case. For $k=2$, as $\alpha(1/2)=\frac{\sqrt{17}-3}{2}\approx 0.5616$, we see that the expected complexity bound is

$$
O\left(\log n + n^{\frac{\sqrt{17}-3}{2}}(\Delta_1+\Delta_2)+n\Delta_1\Delta_2\right).
$$

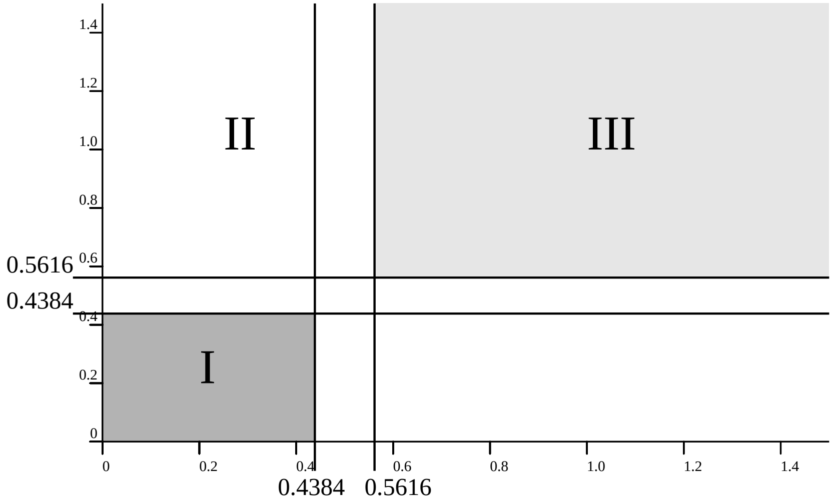

Fig. 4. The complexity regions: in I, the output size dominates. In II, the 1-d complexity term is largest, and III is like point search

The first term accounts for complexity due to search in a tree of height $\log n$. The last term is a volume term, approximately equal to the number of points returned by the query. Both are unavoidable. The middle term is due to complexity related to the perimeter of the query rectangle as a long perimeter cuts many rectangles in the partition. In case $\Delta_1 = 1/n^a$ and $\Delta_2 = 1/n^b$ with $a, b \geq 0$, Figure 4 below shows the regions of the (a, b) plane in which each of the terms dominates. The perimeter term dominates in the white region, the volume term dominates in the dark region, and the search term ($\log n$) dominates in the light region. Point search corresponds to $a = b = \infty$, and a partial match query corresponds to $a = 0, b = \infty$ or vice versa, which falls plainly in the white region. Put differently, we have

$$\mathbf{E}\{N_n\} = \begin{cases} O(\log n) & \text{if } \min(a,b) \geq \alpha(1/2) = \frac{\sqrt{17}-3}{2} \\ O(n^{1-a-b}) & \text{if } \max(a,b) \leq 1 - \alpha(1/2) = \frac{5-\sqrt{17}}{2} \\ O(n^{\frac{\sqrt{17}-3}{2} - \min(a,b)}) & \text{otherwise.} \end{cases}$$

Three-dimensional special case. For $k = 3$, with $\alpha(1/3) \approx 0.7162$ and $\alpha(2/3) \approx 0.3949$, the expected complexity bound is

$$O\left(\log n + n^{0.3949}(\Delta_1 + \Delta_2 + \Delta_3) + n^{0.7162}(\Delta_1\Delta_2 + \Delta_1\Delta_3 + \Delta_2\Delta_3) \right.$$
$$\left. + n\Delta_1\Delta_2\Delta_3\right) ,$$

where we took the liberty of replacing irrational numbers by rational numbers with four significant digits. Note that the two middle terms are the perimeter or lower-dimensional terms. One accounts for the one-dimensional perimeter, and one for the two-dimensional surface area of the query rectangle. Interestingly, there are situations in which each term dominates. For a full picture, let $\Delta_1 = 1/n^a$, $\Delta_2 = 1/n^b$ and $\Delta_3 = 1/n^c$ with $a, b, c \geq 0$. Then we have

$$\mathbf{E}\{N_n\} = \begin{cases} O(\log n) & \text{if } \min(a,b,c) \geq \alpha(2/3) \\ & \qquad = 0.3949 \ (*) \\ O(n^{1-a-b-c}) & \text{if } \max(a,b,c) \leq 1-\alpha(1/3) \\ & \qquad = 0.2838 \ (**) \\ O(n^{0.7162-\min(a,b,c)-(a,b,c)}) & \text{if (*) and (**) fail and} \\ & \text{med}(a,b,c) < \alpha(1/3) \\ & \qquad -\alpha(2/3) = 0.3213 \\ O(n^{0.3949-\min(a,b,c)}) & \text{otherwise.} \end{cases}$$

4 Proof of Theorem 2

In this section, we give a direct probability theoretical proof of the main theorem. By Proposition 1 and the arguments of the previous section, it suffices to prove the following.

Proposition 3 *For fixed s with $0 < s < k$, there exist constants C and C' depending upon s and k only such that in the notation of Proposition 1, for all subsets $S \subseteq \{1, \ldots, k\}$ with $|S| = s$,*

$$C' n^{\alpha(s/k)} \leq \mathbf{E}\left\{\sum_{i=1}^{2n+1} \prod_{j \in S} W_{ij}\right\} \leq C n^{\alpha(s/k)} .$$

Proof. We will prove the upper bound only. The proof uses an embedding argument that constructs an equivalent k-d tree using a different probability model. Assume without loss of generality that the set S consists of the first s coordinates in the rotation (the other cases are not equivalent, but trivially similar). A split along coordinate j will be called a j-split. To determine a split, we just need a uniform $[0,1]$ random variable. So, the construction of the k-d tree may be viewed recursively as follows: at the root, the root rectangle $R_1 = [0,1]^k$ is subjected to a 1-split based on a uniform $[0,1]$ random variable U. One data point is associated with the root (this requires $k-1$ other uniformly distributed coordinates, but they will not be needed for what we need to study), and the sizes of the subtrees associated with the two subrectangles are multinomially distributed with parameters $(n-1, U, 1-U)$. We may apply this procedure recursively, cycling through axes for splitting. After k rounds, thus for rectangles at distance k from the

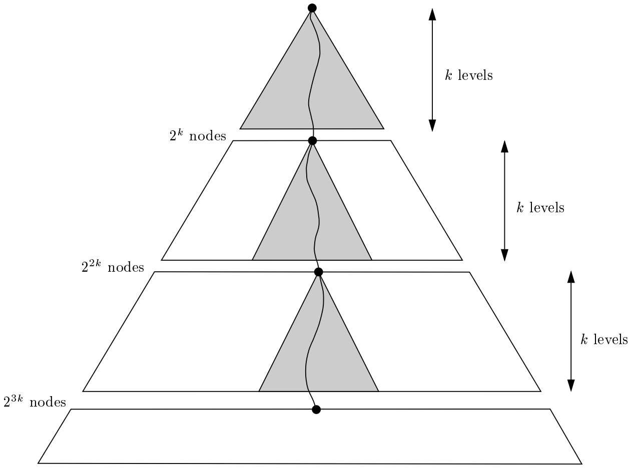

Fig. 5. Tree showing argument in proof of Proposition 3

root, the dimensions of a rectangle are described by a vector $(V_1, \ldots, V_k)$, with independent uniform $[0,1]$ components. As a Binomial(N, p), where N is Binomial(n, q), is Binomial(n, pq), we see that the size of the subtree associated with the rectangle with dimensions $(V_1, \ldots, V_k)$ is stochastically not larger than a Binomial$(n, \prod_{i=1}^k V_i)$ random variable N. If $N = 0$, then the rectangle is either non-existent or a leaf in the final k-d tree. With this mechanism, our tree is an infinite complete binary tree. The actual k-d tree with $2n+1$ rectangles is a subtree of the tree whose nodes represent rectangles R such that $N = \mathrm{Binomial}(n, |R|) > 0$. These N's are dependent, but that will not matter in what follows, by linearity of expectation. We note thus that with each node in the infinite tree, an independent uniform $[0,1]$ random variable is associated, and that the size of a rectangle R whose path from the root to the parent of the rectangle node has uniform random variables $V_1, V_2, \ldots$ is given by

$$(V_1 V_{k+1} V_{2k+1} \ldots, V_2 V_{k+2} V_{2k+2} \ldots, \ldots, V_k V_{2k} V_{3k} \ldots) \ .$$

Returning to the problem at hand, we introduce $V(R)$ and $W(R)$ for a rectangle R at distance ℓ from the root. Here $V(R)$ is the product of all uniforms on that path to the root that correspond to j-splits, $1 \le j \le s$, and $W(R)$ is the product for $s+1 \le j \le k$. Clearly, $|R| = V(R)W(R)$. The quantity of interest to us is

$$\mathbf{E}\left\{\sum_{i=1}^{2n+1} V(R_i)\right\} \le 2\sum_{\ell=0}^{\infty} \mathbf{E}\left\{\sum_{\text{all rectangles } R \text{ at depth } \ell} V(R)\mathbb{1}_{[\mathrm{Binomial}(n,|R|)>0]}\right\} .$$

Here we consider the infinite tree. Leaf nodes in the k-d tree have zero cardinality, but their parents do not. For this reason, we consider only parent nodes, which explains the coefficient 2. Let Z_r and Z'_m represent independent products of r and m independent uniform $[0,1]$ random variables respectively. Then, by looking at levels that are multiples of k only, the last upper bound is not more than $2^{k+1} + 2^{k+1}M$, where

$$M = \sum_{\ell=1}^{\infty} 2^{k\ell}\, \mathbf{E}\left\{ Z_{s\ell} \mathbb{1}_{\left[\text{Binomial}(n, Z_{s\ell} Z'_{(k-s)\ell}) > 0\right]} \right\} .$$

To study M, note first that a uniform $[0,1]$ random variable is distributed as e^{-E}, where E is a standard exponential random variable. Thus, $Z_{s\ell}$ is distributed as $e^{-G_{s\ell}}$, where G_r denotes a Gamma(r) random variable, that is, a random variable with density

$$f(x) = \frac{x^{r-1} e^{-x}}{\Gamma(r)} \ , \ x > 0 \ .$$

Similarly, $Z'_{(k-s)\ell}$ is distributed as $e^{-G_{(k-s)\ell}}$. We will write from now on G and G' for independent gamma random variables. We have then,

$$\begin{aligned} M &\le \sum_{\ell=1}^{\infty} 2^{k\ell}\, \mathbf{E}\left\{ Z_{s\ell} \min\left(1, n Z_{s\ell} Z'_{(k-s)\ell}\right) \right\} \\ &= \sum_{\ell=1}^{\infty} 2^{k\ell} \left(\mathbf{E}\left\{ Z_{s\ell} \mathbb{1}_{\left[n Z_{s\ell} Z'_{(k-s)\ell} \ge 1\right]} \right\} \right. \\ &\quad \left. + \mathbf{E}\left\{ n Z_{s\ell}^2 Z'_{(k-s)\ell} \mathbb{1}_{\left[n Z_{s\ell} Z'_{(k-s)\ell} < 1\right]} \right\} \right) \\ &= \sum_{\ell=1}^{\infty} 2^{k\ell}\, \mathbf{E}\left\{ Z_{s\ell} \mathbb{1}_{\left[n Z_{s\ell} Z'_{(k-s)\ell} \ge 1\right]} \right\} \\ &\quad + \sum_{\ell=1}^{\infty} 2^{k\ell}\, \mathbf{E}\left\{ n Z_{s\ell}^2 Z'_{(k-s)\ell} \mathbb{1}_{\left[n Z_{s\ell} Z'_{(k-s)\ell} < 1\right]} \right\} \\ &= I + II \ . \end{aligned}$$

First we handle I. We have

$$\begin{aligned} I &= \sum_{\ell=1}^{\infty} 2^{k\ell}\, \mathbf{E}\left\{ Z_{s\ell} \mathbb{1}_{\left[n Z_{s\ell} Z'_{(k-s)\ell} \ge 1\right]} \right\} \\ &= \sum_{\ell=1}^{\infty} 2^{k\ell}\, \mathbf{E}\left\{ e^{-G_{s\ell}} \mathbb{1}_{\left[G_{s\ell} + G'_{(k-s)\ell} \le \log n\right]} \right\} \end{aligned}$$

$$= \sum_{\ell=1}^{\infty} 2^{k\ell} \int_{\substack{x+y<\log n \\ x\geq 0, y\geq 0}} e^{-x} \frac{x^{s\ell-1} y^{(k-s)\ell-1}}{\Gamma(s\ell)\Gamma((k-s)\ell)} e^{-x-y}\, dx\, dy$$

$$= \sum_{\ell=1}^{\infty} 2^{k\ell} \int_{0<z<\log n} \int_{0<t<1} z^{k\ell-1} e^{-2tz-(1-t)z} \frac{t^{s\ell-1}(1-t)^{(k-s)\ell-1}}{\Gamma(s\ell)\Gamma((k-s)\ell)}\, dt\, dz$$

(by the transform $x = tz, y = (1-t)z, 0 < t < 1$) .

Similarly, II yields

$$II = \sum_{\ell=1}^{\infty} 2^{k\ell}\, \mathbf{E} \left\{ n Z_{s\ell}^2 Z'_{(k-s)\ell} \mathbb{1}_{\left[n Z_{s\ell} Z'_{(k-s)\ell} < 1 \right]} \right\}$$

$$= \sum_{\ell=1}^{\infty} 2^{k\ell} n \int_{\log n \leq z} \int_{0<t<1} z^{k\ell-1} e^{-3tz-2(1-t)z} \frac{t^{s\ell-1}(1-t)^{(k-s)\ell-1}}{\Gamma(s\ell)\Gamma((k-s)\ell)} dt\, dz$$

so that

$$I + II = \sum_{\ell=1}^{\infty} 2^{k\ell} \int_{0<z<\infty} \int_{0<t<1} z^{k\ell-1} \min(1, ne^{-z}) \times e^{-2tz-(1-t)z} \frac{t^{s\ell}(1-t)^{(k-s)\ell}}{t(1-t)\Gamma(s\ell)\Gamma((k-s)\ell)}\, dt\, dz$$

We first estimate the sum over ℓ, taking only those terms that depend upon ℓ:

$$III = \sum_{\ell=1}^{\infty} \frac{a^{\ell}}{\Gamma(s\ell)\Gamma((k-s)\ell)} ,$$

where $a = 2^k x^s y^{(k-s)}$, and we recall that $x = tz, y = (1-t)z$. Thus,

$$I + II = \int_{0<z<\infty} \int_{0<t<1} III \times \frac{\min(1, ne^{-z})\, e^{-2tz-(1-t)z}}{zt(1-t)}\, dt\, dz .$$

Employing the Stirling approximation

$$\Gamma(\ell) = \left(\frac{\ell}{e}\right)^{\ell} \sqrt{\frac{2\pi}{\ell}} e^{\frac{\theta}{12\ell}}$$

for some $\theta \in [0,1]$ (Whittaker and Watson, 1927, p. 253), we have for $\ell > 0$,

$$\frac{\Gamma(s\ell)\Gamma((k-s)\ell)}{\Gamma(k\ell)} \geq \sqrt{2\pi} e^{-1/12} \sqrt{\frac{k}{s(k-s)\ell}} \left(\frac{s^s(k-s)^{k-s}}{k^k}\right)^{\ell} .$$

Defining $u = s/k$, and

$$\beta = \frac{2x^u y^{1-u}}{u^u(1-u)^{1-u}} = 2z \left(\frac{t}{u}\right)^u \left(\frac{1-t}{1-u}\right)^{1-u} ,$$

we obtain the bound

$$
\begin{aligned}
III &\leq \frac{e^{1/12}\sqrt{s(k-s)}}{\sqrt{2\pi k}} \sum_{\ell=1}^{\infty} \sqrt{\ell} \frac{\left(\frac{a^{1/k}}{u^u(1-u)^{1-u}}\right)^{k\ell}}{\Gamma(k\ell)} \\
&= \frac{e^{1/12}\sqrt{s(k-s)}}{\sqrt{2\pi k}} \sum_{\ell=1}^{\infty} \frac{\sqrt{\ell}\beta^{k\ell}}{\Gamma(k\ell)} \\
&= \frac{e^{1/12}\sqrt{u(1-u)}\, e^{\beta}}{\sqrt{2\pi}} \sum_{\ell=1}^{\infty} \frac{(k\ell)^{\frac{3}{2}}\beta^{k\ell}e^{-\beta}}{(k\ell)!}.
\end{aligned}
$$

We will now show that there is a constant $C_0 > 0$ such that for all $\beta > 0$,

$$
\frac{e^{1/12}e^{\beta}}{\sqrt{2\pi}} \sum_{\ell=1}^{\infty} \frac{(k\ell)^{\frac{3}{2}}\beta^{k\ell}e^{-\beta}}{(k\ell)!} \leq C_0 e^{\beta}\beta^{3/2},
$$

and thus,

$$
III \leq C_0\sqrt{u(1-u)}\, e^{\beta}\beta^{\frac{3}{2}}.
$$

For $\beta > 1$, we have by Jensen's inequality

$$
\begin{aligned}
\frac{e^{1/12}e^{\beta}}{\sqrt{2\pi}} \sum_{\ell=1}^{\infty} \frac{(k\ell)^{\frac{3}{2}}\beta^{k\ell}e^{-\beta}}{(k\ell)!} &\leq \frac{e^{1/12}e^{\beta}}{\sqrt{2\pi}}\, \mathbf{E}\left\{\text{Poisson}^2(\beta)\right\}^{3/4} \\
&\leq \frac{e^{1/12}e^{\beta}}{\sqrt{2\pi}} \left(\beta^2+\beta\right)^{3/4} \leq \frac{e^{1/12}2^{3/4}}{\sqrt{2\pi}} e^{\beta}\beta^{3/2}.
\end{aligned}
$$

For $\beta \leq 1$,

$$
\frac{e^{1/12}e^{\beta}}{\sqrt{2\pi}} \sum_{\ell=1}^{\infty} \frac{(k\ell)^{\frac{3}{2}}\beta^{k\ell}e^{-\beta}}{(k\ell)!} \leq \frac{e^{1/12}e^{\beta}\beta^{k}}{\sqrt{2\pi}} \sum_{j=1}^{\infty} \frac{j^{3/2}}{j!} \leq C^* e^{\beta}\beta^{3/2},
$$

as $\sum_{j=1}^{\infty} \frac{j^{3/2}}{j!}$ converges and $k \geq 2$. Resubstitution yields

$$
\begin{aligned}
& I + II \\
&\quad \leq C_0 \int_0^{\infty}\int_0^1 \sqrt{\frac{zu(1-u)\left(\frac{t}{u}\right)^{3u}\left(\frac{1-t}{1-u}\right)^{3(1-u)}}{t^2(1-t)^2}} \\
&\qquad \times \min(1, ne^{-z})\, e^{z\left(2\left(\frac{t}{u}\right)^u\left(\frac{1-t}{1-u}\right)^{1-u}-t-1\right)}\, dt\, dz \\
&\quad = C\int_0^{\infty} \sqrt{z}\min(1, ne^{-z})\left[\int_0^1 h(t)e^{zg(t)}\, dt\right] dz\ ,
\end{aligned}
$$

where $C = C_0\sqrt{u(1-u)/(u^{3u}(1-u)^{3(1-u)})}$,
$h(t) = t^{3u/2-1}(1-t)^{3(1-u)/2-1}$, and

$$g(t) = 2\left(\frac{t}{u}\right)^u \left(\frac{1-t}{1-u}\right)^{1-u} - t - 1 \, .$$

The behavior of g is easily established: by definition of the Flajolet-Puech function, we have $\sup_{0<t<1} g(t) = \alpha(u)$, and the maximum occurs at $t_0 \in (0,1)$. Furthermore, g is unimodal and locally concave about t_0. Hence, there exists a constant $\nu > 0$ such that $g(t) \le \alpha(u) - \nu(t-t_0)^2$ for all $t \in (0,1)$. Pick $\epsilon > 0$ such that $B = (t_0 - \epsilon, t_0 + \epsilon) \subseteq (0,1)$. Then

$$\begin{aligned}
&\int_{0<t<1} h(t)e^{zg(t)}dt \\
&\quad \le \sup_B h(t) \int_{-\infty}^{\infty} e^{z(\alpha(u)-\nu(t-t_0)^2)}dt + e^{z(\alpha(u)-\nu\epsilon^2)} \int_0^1 h(t)dt \\
&\quad \le \frac{D}{\sqrt{z}} e^{z\alpha(u)} + D' e^{z(\alpha(u)-\nu\epsilon^2)}
\end{aligned}$$

where D and D' are positive constants only depending upon u (through the function h and the constant ν). Resubstitution now yields

$$I + II \le C \int_0^{\infty} \sqrt{z} \min(1, ne^{-z}) \left(\frac{D}{\sqrt{z}} e^{z\alpha(u)} + D' e^{z(\alpha(u)-\nu\epsilon^2)}\right) dz \, .$$

Split the integral over $(0, \log n)$ and $(\log n, \infty)$, and verify that the result is $O(n^{\alpha(u)})$, and that all multiplicative constants indeed only depend upon u and k. □

5 Searching with convex sets

To perform a range search with a convex set C, we may also recursively descend the k-d tree, and visit all subtrees whose root rectangle has a nonempty intersection with C. In this section, to fix the ideas, we will consider $k = 2$ only, although the generalizations to higher dimensions are straightforward. For a fixed convex set C, we let $\mathcal{E}_C$ denote the minimal ellipse containing C. Let the center of $\mathcal{E}_C$ be the origin. Let $\mathcal{E}_C$ have principal axes u and v, with u perpendicular to v. Let R_C be the smallest rectangle aligned with the (u, v) pair that contains $\mathcal{E}_C$ (and thus touches the ellipse in just four points). Let the dimensions of the rectangle R_C (and thus of $\mathcal{E}_C$) in the u and v directions be $\Delta > 0$ and $\Delta' > 0$ respectively. These dimensions are deterministic but may depend on n. A random range search is defined as a range search with convex set $Z + C$, the translation by Z (a uniformly distributed random variable on $[0,1]^k$) of C.

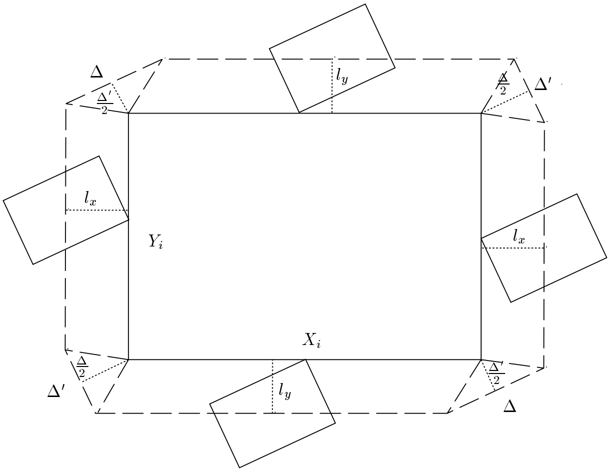

Fig. 6. Areas in Theorem 3

First we generalize Theorem 2 to rotated rectangles. Let Q be a rectangle of size $\Delta \times \Delta'$ parallel to $[0,1]^2$ and centered at the origin. For $\phi \in [0, 2\pi)$, we define Q_ϕ as the rectangle resulting from rotating Q by ϕ about the origin.

Theorem 3 *Let $U_1, \ldots, U_n$ be independent and uniform random variables over $[0,1]^2$, used to construct a 2-d tree, and let $\mathcal{R}_n$ be the partition into rectangles. Let Z be uniformly distributed over $[0,1]^2$, independent of the U_i's, and let N_n be the number of rectangles in $\mathcal{R}_n$ that intersect $Z + Q_\phi$ (and thus the complexity of range search with this set). If Q has dimensions $\Delta \times \Delta'$, then there is a universal constant $\gamma > 0$ (not depending upon n, Δ, Δ' or ϕ), such that*

$$\mathbf{E}\{N_n\} \le \gamma \left(n\Delta\Delta' + (\Delta + \Delta')n^\alpha + \log n \right),$$

where $\alpha = \frac{\sqrt{17}-3}{2}$.

Proof. If a rectangle R_i in $\mathcal{R}_n$ has dimensions $X_i \times Y_i$, then $Z + Q_\phi$ intersects it if and only if Z falls in the octagon outlined in Figure 6, where the tilted rectangles are various positions of the tilted query rectangle. It is easy to see that this octagon in turn is contained in the rectangle R_i extended on top and bottom by l_y (see Figure 6) and on left and right by l_x. Using the same reasoning as in Theorem 2, we note that given $U_1, \ldots, U_n$, the probability that $Z + Q_\phi$ intersects R_i, is bounded by $X_i Y_i + 2\max(l_x, l_y)(X_i + Y_i) +$

$2\Delta\Delta'$. Clearly, $\max(l_x, l_y) \le (\Delta + \Delta')/\sqrt{2}$. Thus,

$$\mathbf{E}\{N_n\} \le \mathbf{E}\left\{\sum_{i=1}^{2n+1}(X_iY_i)\right\} + \sqrt{2}(\Delta+\Delta')\,\mathbf{E}\left\{\sum_{i=1}^{2n+1}(X_i+Y_i)\right\} + (4n+2)\Delta\Delta'$$
$$\le 2H_{n+1} - 1 + c\sqrt{2}(\Delta+\Delta')n^\alpha + (4n+2)\Delta\Delta'$$

for some constant $c > 0$ by Lemma 1 and Proposition 3. Note in particular that the constant c does not depend upon ϕ. □

To prepare for the main result of this section, we will use a fact from classical geometry, stated here in its high-dimensional form. We will use the following result by John (1948).

Lemma 2 *Let S be any bounded set in $\mathbb{R}^k$ not contained in any linear subspace of it. Let $\mathcal{E}_S$ be the smallest ellipsoid containing S (called John's ellipsoid) and $\mathcal{E}'_S$ be the concentric and homothetic ellipsoid at the ratio of $\frac{1}{k}$. Then $\mathcal{E}'_S \subseteq \mathrm{CH}(S) \subseteq \mathcal{E}_S$, where $\mathrm{CH}(S)$ denotes the convex hull of S.*

In particular, John's result implies that $|\mathcal{E}_S| \le k^k|\mathcal{E}'_S| \le k^k|\mathrm{CH}(S)|$. Let $\mathcal{E}$ be an ellipsoid with principal axes of lengths $a_1, \ldots, a_k$, and let B be the unit ball of $\mathbb{R}^k$. Then

$$|\mathcal{E}| = \frac{a_1\cdots a_k}{2^k}|B| = \frac{a_1\cdots a_k}{\Gamma\left(\frac{k+2}{2}\right)}\left(\frac{\sqrt{\pi}}{2}\right)^k .$$

Let S be a set as in the previous lemma, and let $\mathcal{E}_S$ be John's ellipsoid. Assume that $\mathcal{E}_S$ has principal axes of lengths $a_1, \ldots, a_k$. Let R_S be the smallest rectangle whose axes are aligned with those of $\mathcal{E}_S$ that contains $\mathcal{E}_S$ (so that its volume is $a_1 \times \cdots \times a_k$). Then

$$|R_S| \le \left(\frac{2}{\sqrt{\pi}}\right)^k \Gamma\left(\frac{k+2}{2}\right)|\mathrm{CH}(S)| .$$

The main result of this section clearly shows why we call the n^α term the perimeter complexity. In higher dimensions, the complexity of range search involves the volumes of all the lower-dimensional "facets" of C.

Theorem 4 *Let $U_1, \ldots, U_n$ be independent and uniform random variables over $[0,1]^2$, used to construct a 2-d tree, and let $\mathcal{R}_n$ be the partition into rectangles. Let Z be uniformly distributed over $[0,1]^2$, independent of the U_i's, and let N_n be the number of rectangles in $\mathcal{R}_n$ that intersect $Z + C$, where C is a convex set. Then there is a universal constant $\gamma > 0$ (not depending upon n, Δ, Δ' or C), such that*

$$\mathbf{E}\{N_n\} \leq \gamma \left(n \text{ area}(C) + n^{\alpha} \text{ perimeter}(C) + \log n\right),$$

where $\alpha = \frac{\sqrt{17}-3}{2}$.

Proof. Let R_C be the rectangle associated to John's ellipsoid $\mathcal{E}_C$ for C, as defined above. Suppose that it is of size $\Delta \times \Delta'$. Note that the number of comparisons that range search performs with $Z + C$ is not more than that for $Z + R_C$. Therefore, by Theorem 4, for some $\gamma' > 0$,

$$\mathbf{E}\{N_n\} \leq \gamma' \left(n\Delta\Delta' + (\Delta + \Delta')n^{\alpha} + \log n\right).$$

As we noted earlier, $\Delta\Delta' \leq \frac{4}{\pi}$ Area(C). By the convexity of C, and using the small ellipsoid $(1/4)\mathcal{E}_C$ from Lemma 2, we have

$$4\sqrt{\left(\frac{\Delta}{2}\right)^2 + \left(\frac{\Delta'}{2}\right)^2} \leq \text{Perimeter}(C).$$

Also,

$$\text{Perimeter}(R_C) = 2(\Delta+\Delta') \leq 2\sqrt{2}\sqrt{(\Delta)^2 + (\Delta')^2} \leq \sqrt{8}\,\text{Perimeter}(C).$$

Thus we obtain the inequality

$$\mathbf{E}\{N_n\} \leq \gamma' \left((4/\pi)n\text{Area}(C) + \sqrt{8}\,\text{Perimeter}(C)n^{\alpha} + \log n\right). \quad \square$$

6 Local complexities

Our model involved a query rectangle that was centered at a randomly picked point. One may wonder why we did not choose a fixed point. That would have been possible, but reporting the results would have been a nightmare, as we must consider the position of the query rectangle (as a function of n). The answers are indeed affected by border effects. To see this, we indicate why a partial match query in a 2-d tree for all points matching $(*, 0)$ takes expected time $\Theta(n^{\sqrt{2}-1})$ and not $\Theta(n^{0.5616})$ as one would expect by looking at the results of Flajolet and Puech. If the partial match query is for $(*, y_n)$ with y_n going to zero, one must consider the rate with which it tends to zero. There is no room in this paper for this analysis. So, we consider the query $(*, 0)$. Let T_n be the expected time for a 2-d tree of size n. Then, a simple recursive argument shows that roughly speaking,

$$T_n = 1 + 2\,\mathbf{E}\{T_{nU_1U_2}\}$$

where U_1, U_2 are i.i.d. uniform $[0,1]$ random variables. By induction, assuming $T_n \leq Cn^{\gamma}$, we see that γ must be the smallest positive number such that $2/(\gamma+1)^2 \leq 1$. Therefore, $\gamma = \sqrt{2}-1$. In general, with $k-s$ wild cards in $\mathbb{R}^k$, and all coordinates that participate in a partial match query equal to zero, the recursion is of the type

$$T_n = 1 + 2^{k-s}\,\mathbf{E}\left\{T_{nU_1U_2\cdots U_k}\right\} \ .$$

This yields the equation $2^{k-s} = (\gamma+1)^k$. Thus, $\gamma = 2^{1-s/k}-1$, and the expected partial match complexity is $\Theta\left(n^{2^{1-s/k}-1}\right)$.

7 Nearest neighbor search

Consider a random k-d tree as defined above. Let X be a query point uniformly distributed in the unit square. The natural nearest neighbor algorithm referred we will look at is the following. Start with an orthogonal range search with a square box of size $1/n^{1/k}$ centered at X. Repeat with boxes of sizes $k^{t/2}/n^{1/k}$ for $t = 0,1,2,3,\ldots$ until $t+1$, where t is the first nonempty box. Report the nearest point in the $t+1$-st box. We call this algorithm **A**. The purpose of this section is to prove that the expected complexity of this algorithm is $\Theta(n^\rho)$, where

$$\rho = \max_{0\leq s\leq k}\left(\alpha(s/k) - 1 + s/k\right).$$

The constant $\rho \in (0.061, 0.064)$ depends upon k only, and is $(\sqrt{17}-4)/2 \approx 0.0615536$ for $k=2$, is minimal for $k=3$ ($\rho \approx 0.0615254$), and oscillates from that point on. For example, nearest neighbor search in dimensions 2, 4 and 6 have the same expected complexity (as a function of n—the constants may be different), and nearest neighbor search in 3-d is slightly easier than in any other dimension as its ρ-value is smallest! The maximal value for ρ never exceeds 0.064.

We set first the notation we will use. Let $t \geq 1$, we set for all $1 \leq j \leq k$, $\Delta_j = k^{t/2}/n^{1/k}$. Let Q_t be the hypercube with sides all equal to $k^{t/2}/n^{1/k}$, centered at X, a random vector uniformly distributed in $[0,1]^k$, on which an orthogonal range search is performed. Let N_t be the number of data points among $U_1,\ldots,U_n$ falling in Q_t. Let T_t be the complexity of Bentley's orthogonal range search algorithm on Q_t, so that

$$T_t = \sum_{i=1}^{2n+1} \mathbb{1}_{[R_i\cap Q_t\neq\emptyset]} \ ,$$

where R_i is the rectangle in the partition determined by $U_1,\ldots,U_{i-1}$ in which U_i falls. Thus, $R_{n+1},\ldots,R_{2n+1}$ are leaf rectangles. The time taken by algorithm A is

$$T = T_1 + T_2 + \sum_{t\geq 3} T_t \mathbb{1}_{[N_{t-2}=0]} \ .$$

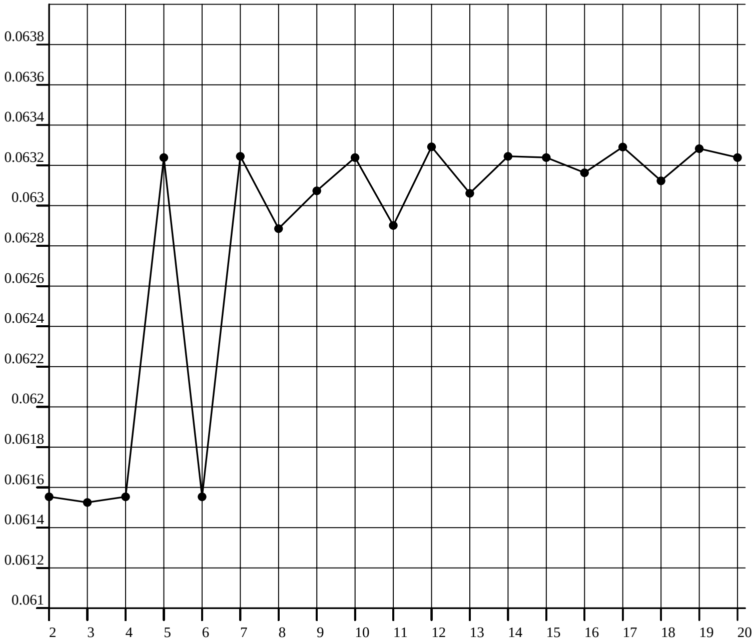

Fig. 7. The function ρ versus k, the dimension. The expected complexity of a natural nearest neighbor algorithm grows as n^ρ

We note that by assumption all points fall in the unit hypercube, and therefore, the maximal index in the last sum cannot exceed $t^* = \lceil 2 \log n/(k \log k) \rceil$.

Fact 1 *Let* $\rho = \max\{\theta(1/k), \theta(2/k), \ldots, \theta((k-1)/k)\}$. *Then there exists a constant* C *not depending upon* t *or* n *such that*

$$\mathbf{E}\{T_t\} \leq C \left(k^{\frac{(k-1)t}{2}} n^\rho + k^{\frac{kt}{2}} \right) .$$

Also,

$$\mathbf{P}\{R_i \cap Q_t \neq \emptyset\} \leq \frac{C}{i} \left(k^{\frac{(k-1)t}{2}} i^\rho + k^{\frac{kt}{2}} \right) .$$

Proof. We apply Theorem 2 with the Δ_j's as given here, and note that

$$\mathbf{E}\{T_t\} \leq C \Big(k^{\frac{kt}{2}} + k^{\frac{(k-1)t}{2}} n^{\theta(1/k)} + k^{\frac{(k-2)t}{2}} n^{\theta(2/k)} + \cdots \\ \cdots + k^{\frac{t}{2}} n^{\theta((k-1)/k)} + \log n \Big) .$$

The first inequality follows immediately from this and the definition of ρ, and the fact that $\log n = o(n^\rho)$. The second inequality uses the fact that

$\mathbf{P}\{R_i \cap Q_t \neq \emptyset\}$ is decreasing in i up to n, and thus, $i\,\mathbf{P}\{R_i \cap Q_t \neq \emptyset\} \leq \mathbf{E}\{T_t\}$ if the sample size used for orthogonal range search is i. The first inequality, with n replaced by i concludes the proof. □

Lemma 3 *Let the following constants be given: $A > 0$, $\gamma > 0$, $\delta > 0$, $\beta \geq 1$, $1 > \rho > 0$, subject to the conditions $A \log \beta \leq 1$, $\log \beta < \delta$. Then*

$$\sum_{t=1}^{\lceil A \log n \rceil} \beta^t \sum_{i=1}^{n} i^{\rho-1} e^{-\gamma(1-i/n)e^{\delta t}} = O(n^\rho) \; .$$

If the conditions are altered so that $\rho = 0$ and $\delta = \log \beta$, then

$$\sum_{t=1}^{\lceil A \log n \rceil} \beta^t \sum_{i=1}^{n} \frac{1}{i} e^{-\gamma(1-i/n)e^{\delta t}} = O(\log n) \; .$$

Proof. It is clear that we may assume without loss of generality that $A \log n$ is integer-valued. Consider first the sum

$$\sum_{t=1}^{\infty} \beta^t e^{-\eta e^{\delta t}}$$

where η will later be replaced by $\gamma(1-i/n)$. By comparison with an integral, we see that this is not more than

$$\beta \int_0^\infty \beta^x e^{-\eta e^{\delta x}} \, dx \; .$$

Set $z = \eta e^{x\delta}$, and verify that the latter expression is smaller than

$$\frac{\beta}{\delta \eta} \int_0^\infty (z/\eta)^{\log \beta/\delta - 1} e^{-z} \, dz \leq \frac{\beta \Gamma(\log \beta/\delta)}{\delta \eta^{\log \beta/\delta}} \; .$$

With this inequality in hand, we note that

$$\sum_{t=1}^{A \log n} \beta^t \sum_{i=1}^{n} i^{\rho-1} e^{-\gamma(1-i/n)e^{\delta t}} \leq n^{\rho-1} \sum_{t=1}^{A \log n} \beta^t \leq \frac{n^{\rho-1} n^{A \log \beta}}{1 - \frac{1}{\beta}} \leq \frac{n^\rho}{1 - \frac{1}{\beta}} \, .$$

Furthermore,

$$\sum_{i=1}^{n-1} i^{\rho-1} \sum_{t=1}^{A \log n} \beta^t e^{-\gamma(1-i/n)e^{\delta t}} \leq \sum_{i=1}^{n-1} i^{\rho-1} \frac{\beta \Gamma(\log \beta/\delta)}{\delta(\gamma(1-i/n))^{\log \beta/\delta}}$$

and thus, it suffices to show that $\sum_{i=1}^{n-1} i^{\rho-1}(1-i/n)^{-b} = O(n^\rho)$, where $b \in [0,1)$. By comparison with an integral, we have

$$\begin{aligned}\sum_{i=1}^{n-1} i^{\rho-1}(1-i/n)^{-b} &= n^\rho \frac{1}{n} \sum_{i=1}^{n-1} (i/n)^{\rho-1}(1-i/n)^{-b} \\ &\le n^\rho \int_0^1 x^{\rho-1}(1-x)^{-b}\, dx \\ &\le B(\rho, 1-b) n^\rho\ ,\end{aligned}$$

where $B(\cdot,\cdot)$ is the beta integral. This concludes the first part of Lemma 3. For the second part, note as before that the contributions in the double sum corresponding to $i = n$ and $i = n-1$ are $O(1)$. For the remainder, we have

$$\begin{aligned}&\sum_{t=1}^{A\log n} \sum_{i=1}^{n-2} i^{-1} \beta^t e^{-\gamma(1-i/n)e^{\delta t}} \\ &\le \sum_{t=1}^{\infty} \beta^t e^{-\gamma(1-1/n)e^{\delta t}} + \sum_{t=1}^{A\log n} \int_1^{n-2} \frac{\beta^t}{x} e^{-\gamma(1-(x+1)/n)e^{\delta t}}\, dx \\ &\le O(1) + \frac{\beta}{\delta\gamma} \int_1^{n-2} \frac{1}{x(1-(x+1)/n)}\, dx \\ &= O(\log n)\ .\end{aligned}$$

This concludes the proof of the second part of Lemma 3. □

Theorem 5 *If T is the time for a nearest neighbor search for algorithm A, then* $\mathbf{E}\{T\} = \Theta(n^\rho)$.

Proof. For the lower bound, we note that $T \ge T_1$, and conclude by the lower bound of Theorem 2 applied to Q_1 and the definition of ρ. For the upper bound, we begin with

$$T = T_1 + T_2 + \sum_{t\ge 3} T_t \mathbb{1}_{[N_{t-2}=0]}\ .$$

Taking expected values, Theorem 2 implies that $\mathbf{E}\{T_1 + T_2\} = O(n^\rho)$. We fix $t \ge 3$ and bound $\mathbf{E}\left\{T_t \mathbb{1}_{[N_{t-2}=0]}\right\}$. The two factors in the expected value are dependent. However, if $N_{t-2,i}$ denotes the number of points among $U_{i+1}, \ldots, U_n$ that fall in Q_{t-2}, then we note that given X, $N_{t-2,i}$ and

$[R_i \cap Q_t \neq \emptyset]$ are independent. Now note that

$$\begin{aligned}
\mathbf{P}\left\{N_{t-2,i}=0|X\right\} &\leq \sup_{x\in[0,1]^k} \mathbf{P}\left\{N_{t-2,i}=0|X=x\right\} \\
&\leq \left(1-\left(\frac{\Delta_{t-2}}{2}\right)^k\right)^{n-i} \\
&\leq \exp\left(-\frac{(n-i)k^{\frac{k(t-2)}{2}}}{2^k n}\right) .
\end{aligned}$$

Thus, as $N_{t-2} \geq N_{t-2,i}$, we have

$$\begin{aligned}
\mathbf{E}\left\{T_t \mathbb{1}_{[N_{t-2}=0]}\right\} &= \mathbf{E}\left\{\sum_{i=1}^{2n+1} \mathbb{1}_{[R_i\cap Q_t\neq\emptyset]}\mathbb{1}_{[N_{t-2}=0]}\right\} \\
&\leq 2\,\mathbf{E}\left\{\sum_{i=1}^{n} \mathbb{1}_{[R_i\cap Q_t\neq\emptyset]}\mathbb{1}_{[N_{t-2,i}=0]}\right\} \\
&= 2\,\mathbf{E}\left\{\sum_{i=1}^{n} \mathbf{P}\left\{R_i\cap Q_t\neq\emptyset|X\right\}\mathbf{P}\left\{N_{t-2,i}=0|X\right\}\right\} \\
&\leq 2\sum_{i=1}^{n}\exp\left(-\frac{(n-i)k^{\frac{k(t-2)}{2}}}{2^k n}\right)\mathbf{E}\left\{\mathbf{P}\left\{R_i\cap Q_t\neq\emptyset|X\right\}\right\} \\
&= 2\sum_{i=1}^{n}\exp\left(-\frac{(n-i)k^{\frac{k(t-2)}{2}}}{2^k n}\right)\mathbf{P}\left\{R_i\cap Q_t\neq\emptyset\right\} \\
&\leq \sum_{i=1}^{n}\exp\left(-\frac{(1-i/n)k^{\frac{k(t-2)}{2}}}{2^k}\right)\frac{2C}{i}\left(k^{\frac{(k-1)t}{2}}i^\rho + k^{\frac{kt}{2}}\right) .
\end{aligned}$$

Thus,

$$\begin{aligned}
\mathbf{E}\left\{T\right\} &\leq O(n^\rho) + \sum_{t=3}^{t^*} 2Ck^{\frac{(k-1)t}{2}}\sum_{i=1}^{n}\exp\left(-\frac{(1-i/n)k^{\frac{k(t-2)}{2}}}{2^k}\right)i^{\rho-1} \\
&\quad + \sum_{t=3}^{t^*} 2Ck^{\frac{kt}{2}}\sum_{i=1}^{n}\frac{1}{i}\exp\left(-\frac{(1-i/n)k^{\frac{k(t-2)}{2}}}{2^k}\right) \\
&= O(n^\rho) + I + II .
\end{aligned}$$

Lemma 3 applies to I if we formally take there $\beta = k^{(k-1)/2}$, $\gamma = 1/(2k)^k$, $A = 2/(k\log k)$, and $\delta = (k\log k)/2$. The conditions of the the first part of Lemma 3, $A\log\beta \leq 1$ and $\log\beta < \delta$, hold, so that $I = O(n^\rho)$. The last part of Lemma 3 applies to II if we set $\beta = k^{k/2}$, $\gamma = 1/(2k)^k$, $A = 2/(k\log k)$, and $\delta = (k \log k)/2 = \log \beta$. Therefore, $II = O(\log n)$. This concludes the proof. □

Bentley conjectured that if one takes a point at random from among the n data points $U_1, \dots, U_n$ in the 2-d tree, its nearest neighbor can be found in $O(1)$ expected time. If the data are put in a $\sqrt{n} \times \sqrt{n}$ regular grid partition of $[0,1]^2$, then each cell would receive on average one data point. It is not hard to see then that the expected time for nearest neighboring searching starting from a given point in a cell takes $O(1)$ expected time. The same is true for all sufficiently regular, dense and rotund partitions, including, for example, the Voronoi diagram or the Delaunay triangulation. If the data are stored in a 2-d tree however, the property fails to hold because of the skinny rectangles. First, to see intuitively what is going on, let X be U_1 and let X' be the nearest neighbor of X among $U_2, \dots, U_n$. Define the nearest neighbor distance $D_n = \|X - X'\|$. Note that D_n is $\Theta(1/\sqrt{n})$ in probability, i.e., $\mathbf{P}\{D_n = o(1/\sqrt{n})\} = o(1)$ and $\mathbf{P}\{D_n = \omega(1/\sqrt{n})\} = o(1)$. For example, for $t > 0$,

$$\mathbf{P}\{\sqrt{n} D_n > t\} \geq \left(1 - \pi t^2/n\right)^n ,$$

so that

$$\begin{aligned}\liminf_{n\to\infty} \mathbf{E}\left\{\sqrt{n} D_n\right\} &\geq \liminf_{n\to\infty} \int_0^\infty (1 - \pi t^2/n)^n \, dt \\ &\geq \int_0^\infty \liminf_{n\to\infty} (1 - \pi t^2/n)^n \, dt \quad \text{(by Fatou's lemma)} \\ &= \int_0^\infty e^{-\pi t^2} \, dt = \frac{1}{2} \, .\end{aligned}$$

This means that a nearest neighbor search for X is roughly equivalent to a $c/\sqrt{n} \times c/\sqrt{n}$ range search. Indeed, just to verify that X' is in fact the claimed nearest neighbor of X, one must at the very least inspect all nodes on rectangle edges that cut the circle S centered at X with radius D_n. Since the rectangles are skinny, the points on the edges may in fact be far from X. Thus a lower bound on the complexity is

$$\mathbf{E}\left\{ \sum_{i=n+1}^{2n+1} \mathbb{1}_{[R_i \cap Q \neq \emptyset]} \right\} ,$$

where Q is the circle of radius D_n centered at X. As D_n is in probability $\Theta(1/\sqrt{n})$, Theorem 1 implies that the expected complexity is $\Omega(n^{\alpha(1/2)-1/2}) \geq \Omega(n^{0.0615})$, thus disproving Bentley's conjecture.

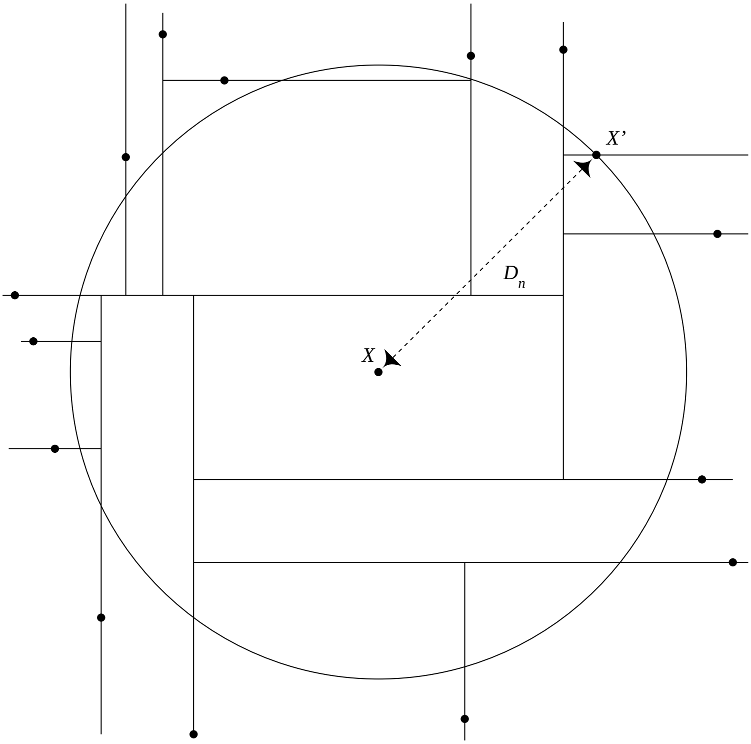

Fig. 8. The nearest neighbor circle is shown. To verify that X' is the nearest neighbor of X, any verification algorithm must examine all points on edges of rectangles that cut the nearest neighbor circle

8 Further work and open problems

Other partitioning algorithms. Our proof method shows the way for the analysis of other partitioning algorithms, such as schemes in which splits are made about medians of $2\ell+1$ elements, as long as the coordinate rotation is respected.

Quadtrees. For quadtree splitting in k dimensions, it is easy to see that not much changes in the analysis, and that in fact Theorem 2 remains valid. This confirms results on partial match queries in random quadtrees by Flajolet, Gonnet, Puech and Robson (1990, 1992).

Expected worst-case complexity. We conjecture that the expected worst case complexity over all range search rectangles of dimensions Δ_j (but with worst-case location of the center) is also bounded from above by the bound given in Theorem 2.

Longest-edge cuts. When we cut rectangles along their longest side, the analysis and the results are very different. The k-d trees are much better behaved, to the point that they are called squarish k-d trees by Devroye, Jabbour, and Zamora (1999).

Non-uniform distributions. Finally, we also intend to study the behavior of k-d trees for nonuniform distributions, although it appears once again that the upper bound of Theorem 2 remains valid for all distributions with a joint density on $[0, 1]^k$.

References

1. P. K. Agarwal: Range searching. In: J. E. Goodman, J. O'Rourke (eds) Handbook of Discrete and Computational Geometry, pp. 575–598. Boca Raton, FL: CRC Press 1997
2. J. L. Bentley: Multidimensional binary search trees used for associative searching. Communications of the ACM **18**, 509–517 (1975)
3. J. L. Bentley: Multidimensional binary search trees in database applications. IEEE Transactions on Software Engineering **SE-5**, 333–340 (1979)
4. J. L. Bentley, J. H. Friedman: Data structures for range searching. ACM Computing Surveys **11**, 397–409 (1979)
5. W. Cunto, G. Lau, P. Flajolet: Analysis of kdt-trees: kd trees improved by local reorganizations. In: Algorithms and Data Structures: Workshop WADS '89 Ottawa, Lecture Notes in Computer Science, vol. 382, pp. 24–38. Berlin: Springer 1989
6. L. Devroye: A note on the height of binary search trees. Journal of the ACM **33**, 489–498 (1986)
7. L. Devroye, J. Jabbour, C. Zamora-Cura: Squarish k-d trees. SIAM Journal of Computing, Accepted for publication, 2000
8. A. Duch, V. Estivill-Castro, C. Martinez: Randomized k-dimensional binary search trees. Technical Report, Departament de Llenguatges i Sistemes Informatics, Polytechnic University of Catalonia, Barcelona, 1998
9. R. A. Finkel, J. L. Bentley: Quad trees: a data structure for retrieval on composite keys. Acta Inf. **4**, 1–9 (1974)
10. P. Flajolet, G. Gonnet, C. Puech, J. M. Robson: Analytic variations on quadtrees. Algorithmica **10**, 473–500 (1992)
11. P. Flajolet, G. Gonnet, C. Puech, M. Robson: The analysis of multidimensional searching in quad-trees. In: Proceedings of the Second Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 100–109. ACM, New York, and SIAM, Philadelphia, 1990
12. P. Flajolet, C. Puech: Partial match retrieval of multidimensional data. Journal of the ACM **33**, 371–407 (1986)
13. D. Gardy, P. Flajolet, C. Puech: Average cost of orthogonal range queries in multiattribute trees. Inf. Syst. **14**, 341–350 (1989)
14. G. H. Gonnet, R. Baeza-Yates: Handbook of Algorithms and Data Structures. Workingham: Addison-Wesley 1991
15. F. John: Extremum problems with inequalities as subsidiary conditions. In: Studies and Essays Presented to R.Courant, pp. 187–204. New York: Interscience 1948
16. D. E. Knuth: The Art of Computer Programming, Vol. 3 : Sorting and Searching (2nd ed). Reading, MA: Addison-Wesley 1997
17. D. T. Lee, C. K. Wong: Worst-case analysis for region and partial region searches in multidimensional binary search trees and quad trees. Acta Inf. **9**, 23–29 (1977)
18. H. M. Mahmoud: Evolution of Random Search Trees. New York: Wiley 1992

19. C. Martinez, A. Panholzer, H. Prodinger: Partial match queries in relaxed multidimensional search trees. Technical Report, Departament de Llenguatges i Sistemes Informatics, Polytechnic University of Catlonia, Barcelona, 1998
20. J. Matoušek: Geometric range searching. ACM Comput. Surv. **26**, 421–461 (1994)
21. R. Neininger, L. Rüschendorf: Limit laws for partial match queries in quadtrees, Technical Report, University of Freiburg, Germany, 1999
22. J. S. Vitter, P. Flajolet: Average-case analysis of algorithms and data structures. In: J. van Leeuwen (ed): Handbook of Theoretical Computer Science, Volume A: Algorithms and Complexity, pp. 431–524. Amsterdam: MIT Press 1990
23. E. T. Whittaker, G. N. Watson: A Course of Modern Analysis. Cambridge, U.K.: Cambridge University Press 1927
24. F. F. Yao: Computational geometry: In: J. van Leeuwen (ed): Handbook of Theoretical Computer Science, Volume A: Algorithms and Complexity, pp. 343–389. Amsterdam: MIT Press 1990