Maintaining range trees in secondary memory Part II: Lower bounds

Michiel H.M. Smid and Mark H. Overmars

RUU-CS-87-21
November 1987

Rijksuniversiteit Utrecht

Vakgroep informatica

Budapestlaan 6 3584 CD Utrecht
Corr. adres: Postbus 80.012 3508 TA Utrecht
Telefoon 030-53 1454
The Netherlands

Maintaining range trees in secondary memory Part II: Lower bounds

Michiel H.M. Smid and Mark H. Overmars

Technical Report RUU-CS-87-21
November 1987

Department of Computer Science
University of Utrecht
P.O.Box 80.012
3508 TA Utrecht
the Netherlands

Maintaining Range Trees in Secondary Memory Part II: Lower Bounds

Michiel H. M. Smid* Mark H. Overmars†

October 1987

Abstract

When storing and maintaining a data structure in secondary memory it is important to partition it into parts such that each query and update passes through a small number of parts. In this way the number of seeks and the amount of data transport required can be kept low. In Part I of this paper a number of partition schemes were given for partitioning range trees. In this paper we will study lower bounds for partitions. In this way we prove that many of the partitions given in Part I are optimal.

1 Introduction

1.1 The partitioning problem

An important searching problem with many applications in e.g. computer graphics and database design is the orthogonal range searching problem.

Definition 1 *Let S be a set of points in d-dimensional space, and let* $([x_1 : y_1], [x_2 : y_2], \ldots, [x_d : y_d])$ *be some hyperrectangle. The* orthogonal range searching problem *asks for all points* $p = (p_1, p_2, \ldots, p_d)$ *in S, such that* $x_1 \leq p_1 \leq y_1, x_2 \leq p_2 \leq y_2, \ldots, x_d \leq p_d \leq y_d$.

Suppose we have a dynamic data structure solving the range searching problem, and suppose this structure is too large to be stored entirely in main memory, a

*Department of Computer Science, University of Amsterdam, Nieuwe Achtergracht 166, 1018 WV Amsterdam, The Netherlands. This author was sponsored by the Netherlands Organisation for the Advancement of Pure Research (ZWO).

†Department of Computer Science, University of Utrecht, P.O.Box 80.012, 3508 TA Utrecht, The Netherlands.

situation that very often occurs in databases. Then this structure has to be stored in secondary memory, and, in order to answer queries and to perform updates, parts of the data structure have to be transported from secondary memory to core, and vice versa. Therefore, it is necessary to partition the data structure into parts, such that a query or an update passes through only a small number of parts (hence only a small amount of data has to be transported). Another situation in which it is useful to partition a data structure, is the case where we want to maintain the structure in secondary memory as a shadow administration (see Smid et al. [4]).

This leads us to the *partitioning problem.*

Definition 2 *A partition of a dynamic data structure, representing a set of n points, is called an $(F(n), G(n))$-partition, if*

1. *Each part has size at most $F(n)$.*
2. *Each update passes through at most $G(n)$ parts.*

In Part I of this paper (cf. [3]) different partition schemes were given for range trees, a data structure solving the range searching problem, obtaining different trade-offs between $F(n)$ and $G(n)$. (In fact, the definition used in [3] was more restricted, requiring that there are at most $O(S(n)/F(n))$ parts, where $S(n)$ is the amount of space required to store the data structure, and requiring that queries pass through a small number of parts also.) In the present paper we study lower bounds for partitions. (All lower bounds for the more general form of partitions we use here, also apply to the more restricted version of [3].) Suppose we have an $(F(n), G(n))$-partition of a range tree. Then given $F(n)$, we are interested in a lower bound for $G(n)$. Similarly, given $G(n)$, we want a lower bound for $F(n)$.

The paper is organized as follows. In Section 1.2, we define range trees (as we use them in this paper), and we give an algorithm to insert and delete points. The notion of range trees we use in this paper is more general than what is normally used. In particular, we do not require trees to be balanced. As the lower bounds we prove apply to any tree (rather than to some trees) in the class of range trees, the bounds also hold under the usual definition. (Moreover, the bounds are more general and would also apply if we for example would use AVL-trees as underlying structure.) In Section 2, we consider one-dimensional range trees. In Section 3, we give lower bounds for two-dimensional range trees, for two types of partitions. The lower bounds we give for the so-called restricted partitions, turn out to be tight, i.e., they match the upper bounds given in [3]. Furthermore, we show that if a two-dimensional range tree is partitioned into parts, such that an update passes through at most k parts, there is a part of size $\Omega(n^{1/k} \log n)$. In Section 4, we generalize the results to the multi-dimensional case. Also here, the lower bounds for restricted partitions are tight. The main result is that for any $(F(n), k)$-

partition of a d-dimensional range tree, $F(n) = \Omega(n^{1/k}(\log n)^{d-1})$. Finally, in Section 5, we give some concluding remarks.

To finish this section, we introduce some notations. First, logarithms, and powers of logarithms, are given in the usual way, i.e., we write $\log n, (\log n)^2$, etc. (in this paper, all logarithms are to the base 2). Furthermore, the k-th iterated logarithm is written as follows. If $k = 1$, then $(\log)^1 n = \log n$. If $k > 1$, then $(\log)^k n = \log\left((\log)^{k-1} n\right)$. The function $\log^* n$ is defined by $\log^* n = \min\{k \geq 1 | (\log)^k n \leq 1\}$.

The data structure we propose to solve the range searching problem, is the range tree. If we have a partition of such a range tree into parts, then the *size* of a part is defined as the number of nodes it contains.

1.2 Range trees

In this section, we define range trees in a very general way.

A *binary tree* is a rooted tree, in which each node has zero or two sons. In this paper, binary trees are used as leaf search trees. That is, if we use a binary tree to represent a set S of real numbers, we store the elements of S in sorted order in the leaves of the tree. Internal nodes of the tree contain information to guide searches. It can be shown by induction on the number of leaves, that a binary tree with n leaves has exactly $2n - 1$ nodes. Binary trees are the building blocks of range trees, which we will define now (cf. Bentley [1], Lueker [2], Willard and Lueker [5]).

Definition 3 *Let S be a set of points in d-dimensional space. A d-*dimensional range tree *T, representing the set S, is defined as follows.*

1. *If $d = 1$, then T is a binary tree, containing the points of S in sorted order in its leaves.*
2. *If $d > 1$, then T consists of a binary tree, called the* main tree, *which contains the points of S in its leaves, ordered according to their first coordinates. Also, each internal node v of this main tree contains an* associated structure, *which is a $(d-1)$-dimensional range tree, representing those points of S which are in the subtree rooted at v, taking only the second to d-th coordinate into account.*

Note that in our definition, we do not impose any balance condition on the binary trees. All results in this paper apply as well for balanced as for unbalanced range trees.

Let T be a d-dimensional range tree, representing the set S, and let v be a node of T (v is a node of the main tree, or of an associated structure, or of an associated structure of an associated structure, etc.). Let S_v be the set of those points of S, which are in the subtree of v. Then node v is said to *represent* the set S_v.

For an algorithm, solving the orthogonal range searching problem using range trees, we refer the reader to [1,2,5]. The update algorithm for these trees is as follows. Suppose we want to insert or delete point p in the range tree. Then we search with p in the main tree to locate its position among the leaves, and we insert or delete p in all associated structures we encounter on our search path. If these associated structures are one-dimensional range trees, we apply the usual insertion/deletion algorithm for binary trees (without rebalancing); otherwise we use the same procedure recursively. Finally, we insert or delete p among the leaves of the main tree.

In this paper, we consider two types of partitions of range trees (see also Part I [3]). A partition of a d-dimensional range tree, where $d > 1$, is called *restricted* if only the main tree is partitioned, whereas associated structures are never subdivided. In a restricted partition, a node of the main tree and its associated structure are contained in the same part. This means that in a restricted partition, there is a part of size $\Omega(n(\log n)^{d-2})$, since the associated structure of the root has size $\Omega(n(\log n)^{d-2})$, as we shall see later. (Note that the size of a part is defined as the number of nodes it contains.) The second type of partitions we consider, are those in which also associated structures are divided into parts. In the rest of this paper, the term partition (without the adjective restricted) indicates a partition of this second type.

2 One-dimensional range trees

In this section, we give a lower bound for partitions of one-dimensional range trees. Note that a one-dimensional range tree is just a binary tree. First, we give a lemma, which will also be used later in the paper.

Lemma 1 *Let T be a binary tree, having at least n leaves. Let S be a subset of the leaves, of cardinality n. Let $m \geq 1$ be a real number. Then the number of nodes in T, representing at least m points of S, is at least $\frac{n}{m} - 1$.*

Proof. The proof is by induction on n. If $1 \leq n < \lceil m \rceil$, then there are no nodes representing at least m points of S, so the number of such nodes is 0, which is at least $\frac{n}{m} - 1$. If $n = \lceil m \rceil$, then the root of T represents at least m points of S. So the total number of nodes in T, representing at least m points of S, is at least 1, which is at least $\frac{n}{m} - 1$. Now let $n > \lceil m \rceil$, and suppose the lemma is proved for smaller values of n. Let v be a node of T representing the entire set S, such that the left son of v represents n_1 points of S, where $1 \leq n_1 \leq n - 1$ (v need not be the root of T, since it is possible that the left son (or the right son) of the root represents the entire set S). By the induction hypothesis, the number of nodes in the left subtree of v, representing at least m points of S, is at least $\frac{n_1}{m} - 1$. Similarly, the right subtree of v contains at least $\frac{n-n_1}{m} - 1$ nodes, representing at least m points of S. Finally, node v itself represents at least m points of S. It follows that the total number of nodes in T, representing at least m points of S, is at least $(\frac{n_1}{m} - 1) + (\frac{n-n_1}{m} - 1) + 1 = \frac{n}{m} - 1$. This proves the lemma. □

This lemma enables us to prove our first lower bound.

Theorem 1 *Let k be a positive integer. Let T be a one-dimensional range tree, representing n points, where $n \geq 2^k$. Suppose the tree T is partitioned into parts, such that each path from the root to a leaf passes through at most k parts. Then there is a part of size at least $\frac{1}{2^{k-1}} n^{1/k}$.*

Proof. The proof is by induction on k. If $k = 1$, the entire tree forms a part in its own. This part has size $2n - 1$, which is at least n. So let $k > 1$, and suppose the theorem is proved for smaller values of k. Let Π be the part of the partition containing the root of T. There are two possibilities.

(i) All nodes in T, representing at least $n^{(k-1)/k}$ points, are contained in part Π. Then, by Lemma 1, this part has size at least $n/n^{(k-1)/k} - 1 = n^{1/k} - 1 \geq \frac{1}{2^{k-1}} n^{1/k}$, since $n \geq 2^k$.

(ii) Otherwise, there is a node v in T, representing at least $n^{(k-1)/k}$ points, which is not contained in part Π. Now consider the subtree of T, having v as its root. The number of points represented by this tree, is at least $n^{(k-1)/k} \geq 2^{k-1}$, since $n \geq 2^k$. We make a new partition of this subtree, by merging part Π and the part containing v together into a new part. The result is a partition of a one-dimensional range tree, representing at least $n^{(k-1)/k}$ points, such that each path from the root to a leaf passes through at most $k - 1$ parts. By the induction hypothesis, there is a part in this new partition of size at least $\frac{1}{2^{k-2}} \left(n^{(k-1)/k}\right)^{1/(k-1)} = \frac{1}{2^{k-2}} n^{1/k}$. Since the sizes of the parts in the new partition are at most twice as large as in our original partition, it follows that the original partition contains a part of size at least $\frac{1}{2^{k-1}} n^{1/k}$. □

In the terminology of Section 1.1, we have proved

Corollary 1 *Let k be a positive integer. For any $(F(n), k)$-partition of a one-dimensional range tree, representing $n \geq 2^k$ points, we have $F(n) \geq \frac{1}{2^{k-1}} n^{1/k}$.*

3 Two-dimensional range trees

In this section, we give lower bounds for both restricted and general partitions of two-dimensional range trees.

3.1 Some preliminary results

We first prove the main lemmas, which will be used in the proofs of the lower bounds.

Lemma 2 *Let $x_0 \geq 1$ be a real number, and let $n \geq 2x_0$ be an integer. Let $f(x)$ be a convex function for $x_0 \leq x \leq n - x_0$. Then for all x, $x_0 \leq x \leq n - x_0$, we have*

$$f(x) + f(n-x) \geq 2\,f(n/2).$$

Proof. Let $x_0 \leq x \leq n - x_0$. Then

$$f(n/2) = f(1/2\,x + 1/2\,(n-x)) \leq 1/2\,f(x) + 1/2\,f(n-x),$$

by the convexity of $f(x)$. □

Lemma 3 *Let $m \geq 1$ be a real number, and let $U(n)$ be a function satisfying*

$$\begin{aligned} U(n) &\geq 0, \quad \text{for } 1 \leq n \leq \lfloor m \rfloor, \\ U(n) &\geq n + \min_{n_1=1,\ldots,n-1}[U(n_1) + U(n-n_1)], \quad \text{for } n \geq \lfloor m \rfloor + 1. \end{aligned}$$

Then $U(n) \geq n \log \frac{n}{m}$, for $n \geq 1$.

Proof. Suppose $1 \leq n \leq \lfloor m \rfloor$. Then $U(n) \geq 0 \geq n \log \frac{n}{m}$, since $\log \frac{n}{m} \leq 0$. So let $n \geq \lfloor m \rfloor + 1$, and suppose the lemma is proved for smaller values of n. Let n_1 be an integer, $1 \leq n_1 \leq n-1$ (note that $n \geq 2$, so such an integer n_1 exists). By the induction hypothesis, we have

$$U(n_1) + U(n-n_1) \geq n_1 \log \frac{n_1}{m} + (n-n_1) \log \frac{n-n_1}{m}.$$

Then, applying Lemma 2, with $f(x) = x \log \frac{x}{m}$, and $x_0 = 1$, we get

$$U(n_1) + U(n-n_1) \geq 2\,\frac{n}{2} \log \frac{n}{2m} = n \log \frac{n}{m} - n.$$

Since n_1 was arbitrary, it follows that

$$U(n) \geq n + \min_{1 \leq n_1 \leq n-1}[U(n_1) + U(n-n_1)] \geq n \log \frac{n}{m}.$$

□

Now we are ready to prove the main lemmas of this section.

Lemma 4 *Let T be a binary tree with n leaves. Let $m \geq 1$ be a real number. For each node v of T, let the weight $wt(v)$ of v be the number of leaves in its subtree. Then*

$$\sum_{v:wt(v)\geq m} wt(v) \geq n \log \frac{n}{m}.$$

Proof. Let $U(n)$ denote the sum $\sum_{v:wt(v)\geq m} wt(v)$. (Strictly speaking we should define $U(n)$ to be the minimum of the expressions $\sum_{v:wt(v)\geq m} wt(v)$ over all binary trees having n leaves.) If $1 \leq n \leq \lfloor m \rfloor$, then of course $U(n) \geq 0$. So let $n \geq \lfloor m \rfloor + 1$. The root of T has weight n, which is at least m. Let n_1 be the number of leaves in the left subtree of the root of T. Then $1 \leq n_1 \leq n-1$, and

$$U(n) \geq n + U(n_1) + U(n-n_1) \geq n + \min_{n_1=1,\ldots,n-1} [U(n_1) + U(n-n_1)].$$

It follows now from Lemma 3, that $U(n) \geq n \log \frac{n}{m}$. □

Corollary 2 *A two-dimensional range tree, representing n points, has size at least $n \log n$.*

Proof. This follows immediately from the above lemma, by taking T the main tree of the range tree, and $m = 1$. □

The next lemma generalizes Lemma 1 to the two-dimensional case.

Lemma 5 *Let T be a two-dimensional range tree, representing at least n points. Let S be a subset of these points, of cardinality n. Let $m \geq 1$ be a real number. Then the total number of nodes in T (in the main tree, or in an associated structure) representing at least m points of S, is at least $\frac{n}{m} \log \frac{n}{m}$.*

Proof. Let $V(n)$ be the number of nodes in T, representing at least m points of S. (Also here we should define $V(n)$ to be the minimum of all these numbers over all range trees and all sets S of cardinality n.) If $1 \leq n \leq \lfloor m \rfloor$, then $V(n) \geq 0$. Let $n \geq \lfloor m \rfloor + 1$. Just as in the proof of Lemma 1, let v be a node in the main tree representing the entire set S, such that the left son of v represents n_1 points of S, where $1 \leq n_1 \leq n-1$. By Lemma 1, the associated structure of v contains at least $\frac{n}{m} - 1$ nodes, representing at least m points of S. Also node v itself represents at least m points of S. Hence

$$\begin{aligned} V(n) &\geq (\frac{n}{m} - 1) + 1 + V(n_1) + V(n-n_1) \\ &\geq \frac{n}{m} + \min_{n_1=1,\ldots,n-1} [V(n_1) + V(n-n_1)]. \end{aligned}$$

It follows that the function $U(n) = m \times V(n)$ satisfies:

$$\begin{aligned} U(n) &\geq 0, \text{ if } 1 \leq n \leq \lfloor m \rfloor, \\ U(n) &\geq n + \min_{n_1=1,\ldots,n-1} [U(n_1) + U(n-n_1)], \text{ if } n \geq \lfloor m \rfloor + 1. \end{aligned}$$

Then by Lemma 3, $U(n) = m \times V(n) \geq n \log \frac{n}{m}$, and hence $V(n) \geq \frac{n}{m} \log \frac{n}{m}$. □

3.2 Lower bounds for restricted partitions

Using Lemma 4 of the preceding section, we are able to give lower bounds for restricted partitions of two-dimensional range trees. Note that in a restricted partition, a node of the main tree and its associated structure are contained in the same part. Hence in a restricted partition of a two-dimensional range tree, there is a part of size at least $2n$ (the part containing the root of the main tree has size at least $1 + (2n - 1) = 2n$). Also, we remind the reader to our notation $(\log)^k n$ for the k-th iterated logarithm, and to the definition of the function $\log^* n$ (see Section 1.1).

Theorem 2 *Consider a two-dimensional range tree, representing n points. Let c be a constant, $c \geq 2$. Suppose the range tree is partitioned into a restricted $(cn, G(n))$-partition. Then $G(n) \geq \frac{1}{\alpha}(1 + \log^* n)$, where $\alpha = 2 + \lceil \log(1 + c) \rceil$.*

Proof. Let T be the main tree. Let $a_1 = 0$, and $a_{i+1} = a_i + c\, i\, 2^{a_i}$ for $i \geq 1$. Let $k = \min\{i \geq 1 | a_{i+1} > \log n\}$. We construct a sequence $v_1, v_2, \ldots, v_k$ of nodes in T, as follows (for each such node v_i, let Π_i be the part of the partition containing v_i). Let v_1 be the root of T. Then v_1 represents at least $n/2^{a_1}$ points. Now let $1 \leq i < k$, and suppose $v_1, \ldots, v_i$ are chosen, in i different parts, such that v_i represents at least $n/2^{a_i}$ points. Consider all nodes in the subtree of T with root v_i, representing at least $n/2^{a_{i+1}}$ points (such nodes exist). These nodes, together with their associated structures, have size at least $\sum_{v:wt(v)\geq m}(1 + wt(v))$, where $wt(v)$ is the number of points represented by node v, $m = n/2^{a_{i+1}}$, and the summation runs over all nodes v in the subtree of T with root v_i having weight at least m. By Lemma 4, this sum is

$$> \left\lceil \frac{n}{2^{a_i}} \right\rceil \log \frac{\lceil n/2^{a_i} \rceil}{n/2^{a_{i+1}}} \geq c\, i\, n.$$

(Since $i < k$, we have $m = n/2^{a_{i+1}} \geq 1$. Hence Lemma 4 can be applied.) Since $|\bigcup_{j=1}^{i} \Pi_j| \leq c\, i\, n$, it follows that there is a node v_{i+1} in the subtree of T with root v_i, representing at least $n/2^{a_{i+1}}$ points, which is not contained in $\bigcup_{j=1}^{i} \Pi_j$.

This procedure gives us nodes $v_1, \ldots, v_k$ in k different parts, such that v_{i+1} is in the subtree of v_i, $i = 1, 2, \ldots, k - 1$. An update of the range tree, which passes through node v_k, passes through at least k parts of the partition. Hence $G(n) \geq k$.

We shall prove now that $k \geq \frac{1}{\alpha}(1+\log^* n)$, where $\alpha = 2+\lceil \log(1+c) \rceil$. Therefore, assume that $k\alpha < 1 + \log^* n$. We show that

$$(\log)^{(1+i\alpha)} n \leq a_{k+1-i}, \quad i = 0, 1, \ldots, k - 1 \tag{1}$$

(note that $(\log)^{(1+i\alpha)} n$ exists for $0 \leq i \leq k - 1$, since $1 + i\alpha \leq 1 + (k - 1)\alpha < 2 - \alpha + \log^* n < \log^* n$).

By definition of k, (1) holds for $i = 0$. Now let $0 \le i \le k-2$, and suppose that $(\log)^{(1+i\alpha)} n \le a_{k+1-i}$. Then we make the following estimations (which are very rough; they lead, however, to the desired result)

$$\begin{aligned}(\log)^{(1+i\alpha)} n &\le a_{k+1-i} \\ &= a_{k-i} + c\,(k-i)\,2^{a_{k-i}} \\ &\le 2^{2a_{k-i}} + c\,(k-i)\,2^{a_{k-i}} \\ &\le 2^{2a_{k-i}} + c\,2^{2a_{k-i}} \qquad \{\text{since } k-i \le 2^{a_{k-i}}\} \\ &= (1+c)\,2^{2a_{k-i}}.\end{aligned}$$

Taking logarithms, we get

$$\begin{aligned}(\log)^{(2+i\alpha)} n &\le \log(1+c) + 2a_{k-i} \\ &\le \lceil \log(1+c) \rceil + 2^{a_{k-i}} \qquad \{\text{since } 2\,a_{k-i} \le 2^{a_{k-i}}\} \\ &= \alpha - 2 + 2^{a_{k-i}}.\end{aligned}$$

Now taking $\alpha - 2$ times the logarithm, and observing that $\log j \le j - 1$, if j is a positive integer, we get $(\log)^{(2+i\alpha+\alpha-2)} n \le 2^{a_{k-i}}$. Hence $(\log)^{(1+(i+1)\alpha)} n \le a_{k-i}$. This proves (1).

Now take $i = k-1$ in (1). Then $(\log)^{(1+(k-1)\alpha)} n \le a_2 = c \le c+1$. Hence $(\log)^{(2+(k-1)\alpha)} n \le \log(1+c) \le \alpha - 2$. Taking $\alpha - 3$ times the logarithm, we get $(\log)^{(2+(k-1)\alpha+\alpha-3)} n = (\log)^{(-1+k\alpha)} n \le 1$. However, since we assumed that $k\alpha < 1 + \log^* n$, we have $(\log)^{(-1+k\alpha)} n > 1$. So we have a contradiction. We have proved that $G(n) \ge k \ge \frac{1}{\alpha}(1 + \log^* n)$. □

Remark. In Part I of this paper [3], it is shown, that there exist two-dimensional range trees, which can be maintained efficiently (that is, they have the same performances as balanced range trees, see Willard and Lueker [5]) , which can be partitioned into a restricted $(O(n), \log^* n + O(1))$-partition. Hence the above lower bound is tight except for constants. In [3] also a restricted $(O(n), 2)$-partition is given. This seems to be in conflict with our lower bound. The reason is that the structure used for obtaining this upper bound is not a real range tree in the sense of our definition but a variation. Hence the lower bound does not apply.

The above theorem gives a lower bound on the number of parts through which an update passes, if all parts have size at most cn. The next theorem considers the opposite point of view: we give a lower bound on the size of parts, if each update passes through at most k parts, where k is a fixed positive integer.

Theorem 3 *Let k be a positive integer. Consider a two-dimensional range tree, representing n points, where $(\log)^k n \ge 16$. Suppose the range tree is partitioned into a restricted $(F(n), k)$-partition. Then $F(n) \ge \frac{1}{2} n (\log)^k n$, i.e., there is a part of size at least $\frac{1}{2} n (\log)^k n$.*

Proof. The proof is by induction on k. If $k = 1$, the entire range tree forms a part in its own. By Corollary 1, this part has size at least $n \log n$. Now let $k > 1$, and suppose the theorem is proved for smaller values of k. Let $m = 4n\frac{(\log)^k n}{(\log)^{k-1} n}$. Let Π be the part of the partition containing the root of the main tree. There are two possibilities.

(i) All nodes in the main tree, representing at least m points, are contained in part Π. Then, by Lemma 4, which may be applied since $m \geq 1$, part Π has size at least

$$\begin{aligned} n \log \frac{n}{m} &= n \log \left(\frac{(\log)^{k-1} n}{4(\log)^k n} \right) \\ &= n \left[(\log)^k n - 2 - (\log)^{k+1} n \right] \\ &\geq 1/2\, n(\log)^k n, \end{aligned}$$

since for $N = (\log)^k n \geq 16$, we have $N - 2 - \log N \geq \frac{1}{2} N$.

(ii) Otherwise, there is a node v in the main tree, representing at least m points, which is not contained in part Π. Consider the subtree (together with its associated structures) having v as its root. Just as in the proof of Theorem 1, we merge part Π and the part containing v together into a new part. This gives us a two-dimensional range tree, representing at least m points. This range tree is partitioned into a restricted $(2F(n), k-1)$-partition. We have

$$\begin{aligned} (\log)^{k-1} m &= (\log)^{k-1} \left(4n \frac{(\log)^k n}{(\log)^{k-1} n} \right) \\ &\geq (\log)^{k-1} \left(\frac{n}{(\log)^{k-1} n} \right) \\ &\geq (\log)^{k-1} \left(\frac{n}{\log n} \right) \\ &\geq 1/2\, (\log)^{k-1} n \\ &\geq 16, \end{aligned}$$

since $(\log)^k n \geq 16$. Here the inequality $(\log)^{k-1} \left(\frac{n}{\log n} \right) \geq 1/2\, (\log)^{k-1} n$ for $k > 1$ can easily be proved by induction on k. Then, by the induction hypothesis, which may be applied, we have

$$\begin{aligned} 2F(n) &\geq 1/2\, m(\log)^{k-1} m \\ &\geq 1/2\, m\, 1/2\, (\log)^{k-1} n \\ &= n \frac{(\log)^k n}{(\log)^{k-1} n} (\log)^{k-1} n \\ &= n(\log)^k n. \end{aligned}$$

Hence $F(n) \geq \frac{1}{2} n(\log)^k n$. This proves the theorem. $\square$

Remark. Also in this case, there exist two-dimensional range trees, having asymptotically the same complexity as balanced range trees (see Willard and Lueker [5]), which can be partitioned into a restricted $(O(n(\log)^k n), k)$-partition. That is, the lower bound of Theorem 3 is tight. See Part I [3].

3.3 A lower bound for general partitions

In Section 2, we proved that if a binary tree is partitioned into parts such that each update passes through at most k parts, there is a part of size $\Omega(n^{1/k})$. Since we saw in Corollary 2, that a two-dimensional range tree requires at least $\log n$ times as much space as a binary tree does, it is to be expected that for an $(F(n), k)$-partition of a two-dimensional range tree, $F(n) = \Omega(n^{1/k} \log n)$. In this section, we show that this is indeed the case. In order to be able to give an inductive proof, we prove a more general result.

Theorem 4 *Let k be a positive integer. Consider a two-dimensional range tree, representing at least n points. Suppose the range tree is partitioned into parts, such that the following holds: There is a node w (in the main tree or in an associated structure), representing at least n points, such that each update which passes through w, passes through at most k parts of the partition.*
Then there is a part of size at least $\frac{1}{2^{k-1}} \frac{1}{k} n^{1/k} \log n$.

Proof. Suppose $k = 1$. Let S be the set of points represented by w. Then $|S| \geq n$. Since $k = 1$, all nodes (either of the main tree, or of an associated structure) representing at least one point of S, are contained in the same part. By Lemma 5, this part has size at least $n \log n$. Let $k > 1$, and suppose the theorem is proved for $k - 1$. Consider a two-dimensional range tree, which satisfies the assumptions of the theorem (for value k). Let S be the set of points represented by node w. Then $|S| \geq n$. Let Π be the part of the partition containing the root of the main tree. We distinguish two cases.

(i) There is a node y in the range tree, representing at least $n^{(k-1)/k}$ points of S, which is not contained in part Π (y may be a node of the main tree, or of an associated structure). Take such a node y. Now take node x in the main tree as follows. If y is a node of the main tree, then $x = y$. Otherwise, y is a node of an associated structure of a node of the main tree. In this case, we let x be this node of the main tree. Consider the subtree with root x, together with its associated structures. We merge part Π and the part containing y together into a new part. This gives us a two-dimensional range tree, representing at least $n^{(k-1)/k}$ points. This range tree is partitioned into parts, such that the following holds. There is a node y, representing at least $n^{(k-1)/k}$ points, such that each update which passes through y, passes through at most $k - 1$ parts of the partition. By the induction hypothesis, there is a part in this new partition, of size at least

$$\frac{1}{2^{k-2}} \frac{1}{k-1} \left(n^{(k-1)/k}\right)^{1/(k-1)} \log \left(n^{(k-1)/k}\right) = \frac{1}{2^{k-2}} \frac{1}{k} n^{1/k} \log n.$$

It follows that in our original partition, there is a part of size at least $\frac{1}{2^{k-1}} \frac{1}{k} n^{1/k} \log n$.

(ii) Otherwise, all nodes y in the range tree, representing at least $n^{(k-1)/k}$ points of S, are contained in part Π. By Lemma 5, which may be applied since $n^{(k-1)/k} \geq 1$, there are at least

$$\frac{n}{n^{(k-1)/k}} \log \left(\frac{n}{n^{(k-1)/k}}\right) = \frac{1}{k} n^{1/k} \log n$$

such nodes y. Hence part Π has size at least

$$\frac{1}{k} n^{1/k} \log n \geq \frac{1}{2^{k-1}} \frac{1}{k} n^{1/k} \log n.$$

This finishes the proof. □

Theorem 5 *Let k be a positive integer. Consider a two-dimensional range tree, representing n points. Suppose the range tree is partitioned into an $(F(n), k)$-partition. Then $F(n) \geq \frac{1}{2^{k-1}} \frac{1}{k} n^{1/k} \log n$, i.e., there is a part of size at least $\frac{1}{2^{k-1}} \frac{1}{k} n^{1/k} \log n$.*

Proof. This follows immediately from Theorem 4, by taking w the root of the main tree. □

4 Multi-dimensional range trees

We shall now generalize the results of Section 3 to the multi-dimensional case. The ideas of the proofs in this section are the same as in Section 3. However, the technical details are a bit more complex.

4.1 Preliminary results

First we give some lemmas, which will be used in the proofs of the lower bounds.

Lemma 6 *Let d be a non-negative integer, and let $h \geq 1$ be a real number. Then*

$$(h-1)^d \geq h^d - d\,h^{d-1}.$$

Proof. Induction on d. □

Lemma 7 *Let $m \geq 1$ be a real number, and let $d \geq 2$ be an integer. Suppose the function $U(n)$ satisfies*

$$\begin{aligned} U(n) &\geq 0, \quad \textit{for} \quad 1 \leq n \leq \lfloor m \rfloor, \\ U(n) &\geq n \left(\log \frac{n}{m}\right)^{d-2} + \min_{n_1=1,\ldots,n-1} [U(n_1) + U(n-n_1)], \quad \textit{for} \quad n \geq \lfloor m \rfloor + 1. \end{aligned}$$

Then $U(n) \geq \frac{1}{d} n \left(\log \frac{n}{m}\right)^{d-1}$, *for* $n \geq \lfloor m \rfloor + 1$.

Proof. Suppose $\lfloor m \rfloor + 1 \leq n \leq 2m$ (note that such an integer n exists). Then $U(n) \geq n \left(\log \frac{n}{m}\right)^{d-2} \geq \frac{1}{d} n \left(\log \frac{n}{m}\right)^{d-1}$, since $0 < \log \frac{n}{m} \leq 1 \leq d$. So let $n > 2m$, and suppose the lemma is proved for smaller values of n. Let $1 \leq n_1 \leq n-1$. Since $n > 2m$, we have $n_1 \geq \lfloor m \rfloor + 1$ or $n - n_1 \geq \lfloor m \rfloor + 1$.

(i) Suppose $n_1 \geq \lfloor m \rfloor + 1$ and $n - n_1 \geq \lfloor m \rfloor + 1$. Then by the induction hypothesis

$$\begin{aligned} & n \left(\log \frac{n}{m}\right)^{d-2} + U(n_1) + U(n-n_1) \\ \geq\ & n \left(\log \frac{n}{m}\right)^{d-2} + \frac{1}{d} n_1 \left(\log \frac{n_1}{m}\right)^{d-1} + \frac{1}{d}(n-n_1) \left(\log \frac{n-n_1}{m}\right)^{d-1} \\ & \qquad \{\text{apply Lemma 2 with } f(x) = x \left(\log \frac{x}{m}\right)^{d-1} \text{ and } x_0 = m\} \\ \geq\ & n \left(\log \frac{n}{m}\right)^{d-2} + \frac{1}{d} n \left(\log \frac{n}{2m}\right)^{d-1} \\ =\ & n \left(\log \frac{n}{m}\right)^{d-2} + \frac{1}{d} n \left(\log \frac{n}{m} - 1\right)^{d-1} \qquad \{\text{apply Lemma 6}\} \\ \geq\ & n \left(\log \frac{n}{m}\right)^{d-2} + \frac{1}{d} n \left[\left(\log \frac{n}{m}\right)^{d-1} - (d-1)\left(\log \frac{n}{m}\right)^{d-2}\right] \\ =\ & \frac{1}{d} n \left(\log \frac{n}{m}\right)^{d-1} + \frac{1}{d} n \left(\log \frac{n}{m}\right)^{d-2} \\ \geq\ & \frac{1}{d} n \left(\log \frac{n}{m}\right)^{d-1}. \end{aligned}$$

(ii) Suppose $n_1 \geq \lfloor m \rfloor + 1$ and $n - n_1 \leq \lfloor m \rfloor$. Then, again by the induction hypothesis,

$$\begin{aligned} & n \left(\log \frac{n}{m}\right)^{d-2} + U(n_1) + U(n-n_1) \\ \geq\ & n \left(\log \frac{n}{m}\right)^{d-2} + \frac{1}{d} n_1 \left(\log \frac{n_1}{m}\right)^{d-1} \qquad \{\text{apply } n_1 \geq n - m\} \\ \geq\ & n \left(\log \frac{n}{m}\right)^{d-2} + \frac{1}{d}(n-m) \left(\log \frac{n_1}{m}\right)^{d-1} \qquad \{\text{apply } \frac{n_1}{m} \geq \frac{n}{2m} \geq 1\} \\ \geq\ & n \left(\log \frac{n}{m}\right)^{d-2} + \frac{1}{d}(n-m) \left(\log \frac{n}{2m}\right)^{d-1} \end{aligned}$$

$$
\begin{aligned}
&= n\left(\log\frac{n}{m}\right)^{d-2}+\frac{1}{d}(n-m)\left(\log\frac{n}{m}-1\right)^{d-1} \qquad \{\text{apply Lemma 6}\}\\
&\geq n\left(\log\frac{n}{m}\right)^{d-2}+\frac{1}{d}(n-m)\left[\left(\log\frac{n}{m}\right)^{d-1}-(d-1)\left(\log\frac{n}{m}\right)^{d-2}\right]\\
&= \frac{1}{d}n\left(\log\frac{n}{m}\right)^{d-1}+\left[\frac{n}{d}+m-\frac{m}{d}-\frac{m}{d}\log\frac{n}{m}\right]\left(\log\frac{n}{m}\right)^{d-2}\\
&\geq \frac{1}{d}n\left(\log\frac{n}{m}\right)^{d-1},
\end{aligned}
$$

since

$$
\frac{n}{d}+m-\frac{m}{d}-\frac{m}{d}\log\frac{n}{m}\geq\frac{n}{d}+m-\frac{m}{d}-\frac{m}{d}\frac{n}{m}=m\left(1-\frac{1}{d}\right)\geq 0.
$$

(iii) Suppose $n_1 \leq \lfloor m\rfloor$ and $n-n_1\geq\lfloor m\rfloor+1$. Then in the same way as in case (ii), we find

$$
n\left(\log\frac{n}{m}\right)^{d-2}+U(n_1)+U(n-n_1)\geq\frac{1}{d}n\left(\log\frac{n}{m}\right)^{d-1}.
$$

It follows from (i), (ii) and (iii), that for $n>2m$

$$
\begin{aligned}
U(n) &\geq n\left(\log\frac{n}{m}\right)^{d-2}+\min_{n_1=1,\ldots,n-1}[U(n_1)+U(n-n_1)]\\
&\geq \frac{1}{d}n\left(\log\frac{n}{m}\right)^{d-1}.
\end{aligned}
$$

□

Lemma 8 *Let $n\geq 1$ and $d\geq 2$ be integers, and let $c\geq\frac{2}{(d-1)!}$ be a real number. Let $a_1=0$, and $a_{i+1}=a_i+c\,d!\,i\,2^{a_i}$ for $i\geq 1$. Let $k=\min\{i\geq 1|a_{i+1}>\log n\}$, and $\alpha=2+\lceil\log(1+c\,d!)\rceil$. Then $k\geq\frac{1}{\alpha}(1+\log^* n)$.*

Proof. Assume that $k\alpha<1+\log^* n$. We show that

$$
(\log)^{(1+i\alpha)}n\leq a_{k+1-i},\ i=0,1,\ldots,k-1 \tag{2}
$$

(note that $(\log)^{(1+i\alpha)}n$ exists for $0\leq i\leq k-1$, since $1+i\alpha\leq 1+(k-1)\alpha< 2-\alpha+\log^* n$).

By definition of k, (1) holds for $i=0$. So let $0\leq i\leq k-2$, and suppose that $(\log)^{(1+i\alpha)}n\leq a_{k+1-i}$. Then, by making the following very rough estimations, we get

$$
\begin{aligned}
(\log)^{(1+i\alpha)}n &\leq a_{k+1-i}\\
&= a_{k-i}+c\,d!\,(k-i)\,2^{a_{k-i}}\\
&\leq 2^{2a_{k-i}}+c\,d!\,(k-i)\,2^{a_{k-i}}\\
&\leq 2^{2a_{k-i}}+c\,d!\,2^{2a_{k-i}} \qquad \{\text{since } k-i\leq 2^{a_{k-i}}\}\\
&= (1+c\,d!)\,2^{2a_{k-i}}.
\end{aligned}
$$

Taking logarithms, we get

$$\begin{aligned}(\log)^{(2+i\alpha)}n &\le \log(1+c\,d!)+2\,a_{k-i}\\ &\le \lceil\log(1+c\,d!)\rceil+2^{a_{k-i}} \qquad \{\text{since } 2\,a_{k-i}\le 2^{a_{k-i}}\}\\ &= \alpha-2+2^{a_{k-i}}.\end{aligned}$$

Now taking $\alpha-2$ times the logarithm, and observing that $\log j \le j-1$, if j is a positive integer, we get $(\log)^{((i+1)\alpha)}n \le 2^{a_{k-i}}$. Hence $(\log)^{(1+(i+1)\alpha)}n \le a_{k-i}$. This proves (1).

Now take $i=k-1$ in (1). Then $(\log)^{(1+(k-1)\alpha)}n \le a_2 = c\,d!$. It follows that $(\log)^{(2+(k-1)\alpha)}n \le \log(c\,d!) \le \alpha-2$. Taking $\alpha-3$ times the logarithm, we get $(\log)^{(-1+k\alpha)}n \le 1$. However, since we assumed that $k\alpha < 1+\log^* n$, we have $(\log)^{(-1+k\alpha)}n > 1$. So we have a contradiction. □

4.2 Lower bounds for restricted partitions

We shall give now two lower bounds for restricted partitions of multi-dimensional range trees. Recall that in a restricted partition, a node of the main tree and its associated structure are contained in the same part.

Lemma 9 *Consider a d-dimensional range tree ($d \ge 2$), representing n points. Let $m \ge 1$ be a real number. For each node v of the main tree, the weight $wt(v)$ of v is defined as the total number of leaves in the associated structure of v (here we count the leaves in the main tree of the associated structure, in associated structures of the associated structure, etc.). Then*

$$\sum_{v:v \text{ represents } \ge m \text{ points}} wt(v) \ge \frac{2}{d!}\,n\,(\log n)^{d-2}\log\frac{n}{m}, \quad \text{if} \quad n \ge \lfloor m\rfloor+1,$$

where the summation runs over all nodes in the main tree, representing at least m points.

Proof. The case $d=2$ is proved already in Lemma 4. So let $d>2$, and suppose the lemma is proved for smaller values of d. Let $V(n)$ denote the sum to be estimated. Then $V(n)\ge 0$ for $1\le n\le\lfloor m\rfloor$. Let $n\ge\lfloor m\rfloor+1$. The root of the main tree represents at least m points. To estimate the weight of the root, we have to count the total number of leaves in its associated structure. By the induction hypothesis (applied for $d-1$ and $m=1$), this weight is at least $\frac{2}{(d-1)!}n(\log n)^{d-2}$. Let n_1 be the number of points represented by the left son of the root of the main tree. Then $1\le n_1\le n-1$. Hence

$$\begin{aligned}V(n) &\ge \frac{2}{(d-1)!}\,n\,(\log n)^{d-2}+V(n_1)+V(n-n_1)\\ &\ge \frac{2}{(d-1)!}\,n\,(\log n)^{d-2}+\min_{1\le n_1\le n-1}[V(n_1)+V(n-n_1)],\end{aligned}$$

for $n \geq \lfloor m \rfloor + 1$. Now let

$$H(n) = 0, \text{ for } 1 \leq n \leq \lfloor m \rfloor,$$
$$H(n) = \frac{(d-1)!}{2} \frac{(\log \frac{n}{m})^{d-2}}{(\log n)^{d-2}} = \frac{(d-1)!}{2}\left(1 - \frac{\log m}{\log n}\right)^{d-2}, \text{ for } n \geq \lfloor m \rfloor + 1.$$

If $1 \leq n \leq \lfloor m \rfloor$, then $H(n)V(n) \geq 0$. Let $n \geq \lfloor m \rfloor + 1$. Then, since $H(n) \geq 0$ and since H is non-decreasing,

$$\begin{aligned} H(n)V(n) &\geq n\left(\log \frac{n}{m}\right)^{d-2} + \min_{1 \leq n_1 \leq n-1}[H(n)V(n_1) + H(n)V(n-n_1)] \\ &\geq n\left(\log \frac{n}{m}\right)^{d-2} + \min_{1 \leq n_1 \leq n-1}[H(n_1)V(n_1) + H(n-n_1)V(n-n_1)]. \end{aligned}$$

It follows from Lemma 7, that

$$H(n)V(n) \geq \frac{1}{d} n \left(\log \frac{n}{m}\right)^{d-1}, \text{ if } n \geq \lfloor m \rfloor + 1,$$

and hence

$$V(n) \geq \frac{2}{d!} n (\log n)^{d-2} \log \frac{n}{m}, \text{ if } n \geq \lfloor m \rfloor + 1. \quad \square$$

Now apply this lemma, with $m = 1$. Then we get

Corollary 3 *A d-dimensional range tree ($d \geq 2$), representing n points, has size at least* $\frac{2}{d!} n (\log n)^{d-1}$.

Lemma 9 enables us to prove lower bounds for restricted partitions. Note that it follows from Corollary 3, that in a restricted partition of a d-dimensional range tree, there is a part of size at least $\frac{2}{(d-1)!} n (\log n)^{d-2}$.

Theorem 6 *Consider a d-dimensional range tree ($d \geq 2$), representing n points, where $(d-2)\log\log n \leq \frac{1}{2}\log n$. Let c be a constant, $c \geq \frac{2}{(d-1)!}$. Suppose the range tree is partitioned into a restricted $(c\, n\, (\log n)^{d-2}, G(n))$-partition. Then $G(n) \geq \frac{1}{\alpha}(1 + \log^* n)$, where $\alpha = 2 + \lceil \log(1 + c\, d!) \rceil$.*

Proof. Let T be the main tree. Let $a_1 = 0$, and $a_{i+1} = a_i + c\, d!\, i\, 2^{a_i}$, for $i \geq 1$. Let $k = \min\{i \geq 1 | a_{i+1} > \log n\}$. We construct a sequence $v_1, v_2, \ldots, v_k$ of nodes in T, as follows (for each such node v_i, let Π_i be the part of the partition containing v_i). Let v_1 be the root of T. Then v_1 represents at least $n/2^{a_1}$ points. Now let $1 \leq i < k$, and suppose $v_1, \ldots, v_i$ are chosen, in i different parts, such that v_i represents at least $n/2^{a_i}$ points. Consider all nodes in the subtree of T with root v_i, representing at least $n/2^{a_{i+1}}$ points. These nodes, together with their associated structures, have size at least $\sum_{v:v \text{ represents } \geq m \text{ points}}(1 + wt(v))$. Here $wt(v)$ is the total number of leaves in the associated structure of node v, $m = n/2^{a_{i+1}}$, and the summation runs over all nodes v in the subtree of T with root v_i, representing at least m points. By Lemma 9, this sum is

$$\begin{aligned}
&> \frac{2}{d!}\left\lceil\frac{n}{2^{a_i}}\right\rceil\left(\log\left\lceil\frac{n}{2^{a_i}}\right\rceil\right)^{d-2}\log\frac{\lceil n/2^{a_i}\rceil}{n/2^{a_{i+1}}}\\
&\geq \frac{2}{d!}\frac{n}{2^{a_i}}\left(\log\frac{n}{2^{a_i}}\right)^{d-2} c\, d!\, i\, 2^{a_i}\\
&= 2\,c\,i\,n\left(\log\frac{n}{2^{a_i}}\right)^{d-2}\\
&\geq 2\,c\,i\,n\left(\log\frac{n}{\log n}\right)^{d-2} \quad \{\text{since } \log n \geq a_{i+1} \geq 2^{a_i}\}\\
&= 2\,c\,i\,n\,(\log n - \log\log n)^{d-2}\\
&\geq c\,i\,n\,(\log n)^{d-2}.
\end{aligned}$$

Here the last inequality follows from Lemma 6, and the assumption that $(d-2)\log\log n \leq \frac{1}{2}\log n$: $\left(\frac{\log n}{\log\log n}-1\right)^{d-2} \geq \left(\frac{\log n}{\log\log n}\right)^{d-2} - (d-2)\left(\frac{\log n}{\log\log n}\right)^{d-3} \geq \frac{1}{2}\left(\frac{\log n}{\log\log n}\right)^{d-2}$.
(Note that since $1 \leq i < k$, we have $m = n/2^{a_{i+1}} \geq 1$ and

$$\lfloor m\rfloor + 1 \leq m+1 \leq \frac{n}{2^{a_2}} + 1 = \frac{n}{2^{c\,d!}} + 1 \leq \frac{n}{16} + 1 \leq n.$$

Hence Lemma 9 can be applied.)

Now since $|\bigcup_{j=1}^{i}\Pi_j| \leq c\,i\,n\,(\log n)^{d-2}$, it follows that there is a node v_{i+1} in the subtree of T with root v_i, representing at least $n/2^{a_{i+1}}$ points, which is not contained in $\bigcup_{j=1}^{i}\Pi_j$.

This procedure gives us nodes $v_1,\ldots,v_k$ in k different parts, such that v_{i+1} is in the subtree of $v_i, i = 1,2,\ldots,k-1$. An update of the range tree, which passes through node v_k, passes through at least k parts of the partition. Hence $G(n) \geq k$. Then it follows from Lemma 8, that $g(n) \geq \frac{1}{\alpha}(1+\log^* n)$. □

Remark. In Part I [3], it is shown, that there exist d-dimensional range trees (where $d \geq 2$), having the same complexity as balanced range trees (see [5]), which can be partitioned into a restricted $(O(n(\log n)^{d-2}), \log^* n + O(1))$-partition. Hence the lower bound of Theorem 6 is tight.

In the next theorem we consider the opposite point of view: We proof a lower bound on the sizes of parts, if each update passes through at most k parts, for a fixed positive integer k.

Theorem 7 *Let k be a positive integer. Consider a d-dimensional range tree ($d \geq 2$), representing n points, where $(\log)^k n \geq 16$. Suppose the range tree is partitioned* *into a restricted $(F(n), k)$-partition. Then $F(n) \geq \left(\frac{1}{2}\right)^{(d-2)(k-1)} \frac{1}{d!} n(\log n)^{d-2} (\log)^k n$, that is, there is a part of size at least*

$$\left(\frac{1}{2}\right)^{(d-2)(k-1)} \frac{1}{d!} n(\log n)^{d-2} (\log)^k n.$$

Proof. The proof is by induction on k. If $k = 1$, the entire range tree forms a part in its own. By Corollary 3, this part has size at least $\frac{2}{d!} n(\log n)^{d-1}$. Now let $k > 1$, and suppose the theorem is proved for smaller values of k. Let $m = 4n \frac{(\log)^k n}{(\log)^{k-1} n}$. Let Π be the part of the partition, containing the root of the main tree. There are two possibilities.

(i) All nodes in the main tree, representing at least m points, are contained in part Π. Then, by Lemma 9, part Π has size at least

$$\begin{aligned} \frac{2}{d!} n(\log n)^{d-2} \log \frac{n}{m} &= \frac{2}{d!} n(\log n)^{d-2} \left[(\log)^k n - 2 - (\log)^{k+1} n\right] \\ &\geq \frac{1}{d!} n(\log n)^{d-2} (\log)^k n \\ &\geq \left(\frac{1}{2}\right)^{(d-2)(k-1)} \frac{1}{d!} n(\log n)^{d-2} (\log)^k n. \end{aligned}$$

Here the first inequality follows from the fact that for $N = (\log)^k n \geq 16$, we have $N - 2 - \log N \geq \frac{1}{2} N$.
(Note that $m \geq \frac{n}{(\log)^{k-1} n} \geq 1$, and that $\lfloor m \rfloor + 1 \leq 2m \leq n$, since for $N = (\log)^k n \geq 16$, we have $8\,N \leq 2^N$. Hence Lemma 9 can be applied.)

(ii) Otherwise, there is a node v in the main tree, representing at least m points, which is not contained in part Π. Consider the subtree (together with its associated structures) having v as its root. We merge part Π and the part containing v together into a new part. This gives us a d-dimensional range tree, representing at least m points. This range tree is partitioned into a restricted $(2F(n), k-1)$-partition. We have

$$\begin{aligned} (\log)^{k-1} m &= (\log)^{k-1} \left(4n \frac{(\log)^k n}{(\log)^{k-1} n}\right) \\ &\geq (\log)^{k-1} \left(\frac{n}{(\log)^{k-1} n}\right) \\ &\geq (\log)^{k-1} \left(\frac{n}{\log n}\right) \\ &\geq 1/2\,(\log)^{k-1} n. \end{aligned}$$

Here the inequality $(\log)^{k-1} \left(\frac{n}{\log n}\right) \geq 1/2\,(\log)^{k-1} n$, for $k > 1$ and $(\log)^k n \geq 16$ can easily be proved by induction on k. If we substitute $k = 2$ in the above inequalities, we get

$$(\log m)^{d-2} \geq \left(\frac{1}{2}\right)^{d-2} (\log n)^{d-2}.$$

Now apply the induction hypothesis (note that this is allowed since $(\log)^{k-1}m \geq 1/2\,(\log)^{k-1}n \geq 16$). Then

$$\begin{aligned} 2\,F(n) &\geq \left(\frac{1}{2}\right)^{(d-2)(k-2)} \frac{1}{d!} m(\log m)^{d-2}(\log)^{k-1}m \\ &\geq \left(\frac{1}{2}\right)^{(d-2)(k-2)} \frac{1}{d!} m \left(\frac{1}{2}\right)^{d-2} (\log n)^{d-2}\, 1/2\, (\log)^{k-1}n \\ &= \left(\frac{1}{2}\right)^{(d-2)(k-1)} \frac{1}{d!} 2\, n(\log n)^{d-2}(\log)^{k}n. \end{aligned}$$

Hence $F(n) \geq \left(\frac{1}{2}\right)^{(d-2)(k-1)} \frac{1}{d!} n(\log n)^{d-2}(\log)^{k}n$. This proves the theorem. □

Remark. Also in this case, there exist d-dimensional range trees, having asymptotically the same complexity as balanced range trees, which can be partitioned into a restricted $(O(n(\log n)^{d-2}(\log)^{k}n), k)$-partition. Hence also this lower bound is tight. See Part I [3].

4.3 A lower bound for general partitions

We shall now generalize the lower bounds of Theorems 1 and 5. Just as in the two-dimensional case, we prove a more general result. First, we prove the following lemma.

Lemma 10 *Consider a d-dimensional range tree ($d \geq 2$), representing at least n points. Let S be a subset of these points, of cardinality n. Let $m \geq 1$ be a real number. Then the total number of nodes in the range tree (in the main tree, or in an associated structure, or in an associated structure of an associated structure, etc.), representing at least m points of S, is at least $\frac{2}{d!}\frac{n}{m}\left(\log \frac{n}{m}\right)^{d-1}$, if $n \geq \lfloor m \rfloor + 1$.*

Proof. For $d = 2$, the claim follows from Lemma 5. Let $d > 2$, and suppose the lemma is proved for smaller values of d. Let $V(n)$ be the total number of nodes in the range tree, representing at least m points of S. If $1 \leq n \leq \lfloor m \rfloor$, then $V(n) \geq 0$. Let $n \geq \lfloor m \rfloor + 1$. Let v be a node in the main tree, representing the entire set S, such that the left son of v represents n_1 points of S, where $1 \leq n_1 \leq n-1$ (v need not be the root of the main tree, since it is possible that the left son (or the right son) of the root represents the entire set S). By the induction hypothesis, the associated structure of v contains at least $\frac{2}{(d-1)!}\frac{n}{m}\left(\log \frac{n}{m}\right)^{d-2}$ nodes, representing at least m points of S. Hence

$$V(n) \geq \frac{2}{(d-1)!}\frac{n}{m}\left(\log \frac{n}{m}\right)^{d-2} + \min_{1 \leq n_1 \leq n-1} [V(n_1) + V(n-n_1)],$$

for $n \geq \lfloor m \rfloor + 1$. Then it follows from Lemma 7, that

$$V(n) \geq \frac{2}{d!}\frac{n}{m}\left(\log \frac{n}{m}\right)^{d-1}, \quad \text{for } n \geq \lfloor m \rfloor + 1.$$

□

Theorem 8 *Let k be a positive integer. Consider a d-dimensional range tree, representing at least n points, where $n \geq 2^k$. Suppose the range tree is partitioned into parts, such that the following holds. There is a node w (in the main tree, or in an associated structure, or in an associated structure of an associated structure, etc.), representing at least n points, such that each update which passes through w passes through at most k parts of the partition.*
Then there is a part of size at least $\frac{2}{d!}\frac{1}{2^{k-1}}\left(\frac{1}{k}\right)^{d-1} n^{1/k} (\log n)^{d-1}$.

Proof. Suppose $k = 1$. Let S be the set of points represented by w. Then $|S| \geq n$. Since $k = 1$, all nodes representing at least one point of S, are contained in the same part. By Lemma 10, with $m = 1$, this part has size at least $\frac{2}{d!}n(\log n)^{d-1}$. Let $k > 1$, and suppose the theorem is proved for $k - 1$. Consider a d-dimensional range tree, which satisfies the assumptions of the theorem (for value k). Let S be the set of points represented by node w. Then $|S| \geq n$. Let Π be the part of the partition containing the root of the main tree. We distinguish two cases.

(i) There is a node y in the range tree, representing at least $n^{(k-1)/k}$ points of S, which is not contained in part Π (y may be a node of the main tree, or of an associated structure, or of an associated structure of an associated structure, etc.). Take such a node y. Now take node x in the main tree as follows. If y is a node of the main tree, then $x = y$. Otherwise, y is contained in an associated structure of a node in the main tree. In this case, we let x be this node of the main tree. Consider the subtree with root x, together with its associated structures. We merge part Π and the part containing y together into a new part. This gives us a d-dimensional range tree, representing at least $n^{(k-1)/k}$ points, where $n^{(k-1)/k} \geq 2^{k-1}$. This range tree is partitioned into parts, such that the following holds. There is a node y, representing at least $n^{(k-1)/k}$ points, such that each update which passes through y, passes through at most $k-1$ parts of the partition. By the induction hypothesis, which may be applied, there is a part in this new partition, of size at least

$$\frac{2}{d!}\frac{1}{2^{k-2}}\left(\frac{1}{k-1}\right)^{d-1}\left(n^{(k-1)/k}\right)^{1/(k-1)}\left(\log\left(n^{(k-1)/k}\right)\right)^{d-1}$$
$$= \frac{2}{d!}\frac{1}{2^{k-2}}\left(\frac{1}{k}\right)^{d-1} n^{1/k} (\log n)^{d-1}.$$

It follows that in our original partition, there is a part of size at least

$$\frac{2}{d!}\frac{1}{2^{k-1}}\left(\frac{1}{k}\right)^{d-1} n^{1/k} (\log n)^{d-1}.$$

(ii) Otherwise, all nodes y in the range tree, representing at least $n^{(k-1)/k}$ points of S, are contained in part Π. By Lemma 10, there are at least

$$\frac{2}{d!}\frac{n}{n^{(k-1)/k}}\left(\log\left(\frac{n}{n^{(k-1)/k}}\right)\right)^{d-1} = \frac{2}{d!}\left(\frac{1}{k}\right)^{d-1} n^{1/k} (\log n)^{d-1}$$

such nodes y.
(Note that $n^{(k-1)/k} \geq 1$, and since $n \geq 2^k$, we have $n \geq 2\,n^{(k-1)/k} \geq \lfloor n^{(k-1)/k} \rfloor + 1$. Hence Lemma 10 can be applied.)
It follows that part II has size at least

$$\frac{2}{d!}\left(\frac{1}{k}\right)^{d-1} n^{1/k}\,(\log n)^{d-1} \geq \frac{2}{d!}\frac{1}{2^{k-1}}\left(\frac{1}{k}\right)^{d-1} n^{1/k}\,(\log n)^{d-1}.$$

This finishes the proof. □

Theorem 9 *Let k be a positive integer. Consider a d-dimensional range tree ($d \geq 2$), representing n points, where $n \geq 2^k$. Suppose the range tree is partitioned into an $(F(n), k)$-partition. Then*

$$F(n) \geq \frac{2}{d!}\frac{1}{2^{k-1}}\left(\frac{1}{k}\right)^{d-1} n^{1/k}\,(\log n)^{d-1}.$$

Proof. This follows immediately from Theorem 8, by taking w the root of the main tree. □

5 Concluding remarks

This paper is the second part in a series of two, in which we studied the following problem. Given a partition of a range tree into parts of size at most $F(n)$, such that each update passes through at most $G(n)$ of these parts, what is the relation between $F(n)$ and $G(n)$. This is useful if we want to store and maintain the range tree in secondary memory. In that case the number of seeks to perform an update is at most $G(n)$, and the total amount of data that has to transported from secondary memory to main memory, and vice versa, is bounded above by $F(n)G(n)$. In Part I [3], we have given several partition schemes for range trees. In the present paper, we have studied lower bounds for the partitioning of both two- and multi-dimensional range trees, in the following sense. Given $F(n)$ (resp. $G(n)$), derive lower bounds on $G(n)$ (resp. $F(n)$). We considered two types of partitions. In a restricted partition, a node of the main tree is contained in the same part as its associated structure. We showed e.g. in the two-dimensional case, that if $G(n) = k$, then $F(n) \geq \frac{1}{2}n(\log)^k n$. All lower bounds for restricted partitions we have given, turn out to be tight. In a general partition, also associated structures are partitioned. It was shown e.g. that if $G(n) = k$, then $F(n) \geq \frac{1}{2^{k-1}}\frac{1}{k}n^{1/k}\log n$ (again in the two-dimensional case). Notice that in the above examples, it is crucial that k is fixed, i.e., does not depend on n (the proofs are by induction on k).

Acknowledgements

We want to thank Leen Torenvliet and Peter van Emde Boas, for reading preliminary versions of this paper, and for discussing the results with us. Furthermore, we want to thank Leen Torenvliet for the technical assistance during the preparation of this paper.

References

[1] J.L. Bentley. *Decomposable Searching Problems.* Inform. Proc. Lett. **8** (1979), pp. 244-251.

[2] G.S. Lueker. *A Data Structure for Orthogonal Range Queries.* Proc. 19-th Annual IEEE Symp. on Foundations of Computer Science, 1978, pp. 28-34.

[3] M.H. Overmars, M.H.M. Smid, M.T. de Berg and M.J. van Kreveld. *Maintaining Range Trees in Secondary Memory, Part I: Partitions.* Report FVI-87-14, University of Amsterdam, 1987.

[4] M.H.M. Smid, L. Torenvliet, P. van Emde Boas and M.H. Overmars. *Two Models for the Reconstruction Problem for Dynamic Data Structures.* Report FVI-87-13, University of Amsterdam, 1987.

[5] D.E. Willard and G.S. Lueker. *Adding Range Restriction Capability to Dynamic Data Structures.* Journal of the ACM **32** (1985), pp. 597-617.