

# Median and Hybrid Median $K$ -Dimensional Trees<sup>\*</sup>

Amalia Duch<sup>1</sup>[0000–0003–4371–1286], Conrado Martínez<sup>1</sup>[0000–0003–1302–9067] ,  
Mercè Pons<sup>2</sup>, and Salvador Roura<sup>1</sup>[0000–0003–4394–5939]

<sup>1</sup> Dept. of Computer Science. Universitat Politècnica de Catalunya.  
{`duch, conrado, roura`}@`cs.upc.edu`

<sup>2</sup> Departament d'Ensenyament. Generalitat de Catalunya. `mercep@gmail.com`

**Abstract.** We consider here two new variants of  $K$ -dimensional binary search trees ( $K$ -d trees): median  $K$ -d trees and hybrid-median  $K$ -d trees. These two kinds of trees are designed with the aim to get a tree as balanced as possible. This goal is attained by heuristics that choose for each node of the  $K$ -d tree the appropriate coordinate to discriminate. In the case of median  $K$ -d trees, the chosen dimension to discriminate at each node is the one whose point value at that node is the most centered one. In hybrid-median  $K$ -d trees, the heuristic is similar except that it should be followed in a cyclic way, meaning that, at every path of the tree, no dimension can be re-selected to discriminate unless all the other dimensions have already been selected. We study the expected internal path length (IPL) and the expected cost of random partial match (PM) searches in both variants of  $K$ -d trees. For both variants, we prove that the expected IPL is of the form  $c_K \cdot n \log_2 n + \text{lower order terms}$ , and the expected cost of PM is of the form  $\Theta(n^\alpha)$  with  $\alpha = \alpha(s, K)$ . We give the explicit equations satisfied by the constants  $c_K$  and the exponents  $\alpha$  which we can then numerically solve. Moreover, we prove that as  $K \rightarrow \infty$  the trees in both variants tend to be perfectly balanced ( $c_K \rightarrow 1$ ) and we also show that  $\alpha \rightarrow \log_2(2 - s/K)$  for median  $K$ -d trees when  $K \rightarrow \infty$ . In the case of hybrid median  $K$ -d trees we conjecture that  $\alpha \rightarrow 1 - s/K$ , when  $K \rightarrow \infty$ , which would be optimal.

**Keywords:**  $K$ -d trees · Multidimensional data structures · Partial match queries · Analysis of algorithms.

## 1 Introduction

In this work we study two variants of  $K$ -dimensional binary search trees [1, 14] ( $K$ -d trees, for short): *median  $K$ -d trees* and *hybrid median  $K$ -d trees*; both were introduced by Pons [12] in 2010. When built from uniformly distributed input

---

<sup>\*</sup> This work has been supported by funds from the MOTION Project (Project PID2020-112581GB-C21) of the Spanish Ministry of Science & Innovation MCIN/AEI/10.13039/501100011033.

data sets, these two simple variants of  $K$ -d trees achieve better costs for exact searches and insertions than other variants of  $K$ -d trees. They also perform better with respect to *partial match queries* which in turn implies better performance in other *associative queries* like *orthogonal range* or *nearest neighbour* queries.

Recall that a  $K$ -d tree is a binary search tree that stores a collection  $F$  of items, each endowed with a  $K$ -dimensional key  $\mathbf{x} = (x_0, \dots, x_{K-1})$ . In addition to the data point key  $\mathbf{x}$ , each node  $\langle \mathbf{x}, j \rangle$  of a  $K$ -d tree stores a *discriminant*, a value  $j$ ,  $0 \leq j < K$ , which is the coordinate that will be used to split the inserted keys into the left and right subtrees rooted at  $\langle \mathbf{x}, j \rangle$ : the data points with a key  $\mathbf{y}$  such that  $y_j < x_j$  are recursively inserted into the left subtree, whereas those with a key  $\mathbf{z}$  such that  $x_j < z_j$  are recursively inserted into the right one<sup>3</sup>.

The original  $K$ -d trees —we will refer to these as *standard  $K$ -d trees*— were introduced by Bentley in the mid 70s [1] with a rule that assigns discriminants in a cyclic way. Thus, a node at level  $\ell \geq 0$  has discriminant  $\ell \bmod K$ . Several variants of  $K$ -d trees differ in the way in which the discriminants are assigned to nodes, whereas other variants apply local (for example, *Kdt trees* [2]) or global rebalancing rules (for example, *divided  $K$ -d trees* [8]). Among the variants that use alternative rules to assign discriminants we have *relaxed  $K$ -d trees* [4], which assign discriminants uniformly and independently at random, and *squarish  $K$ -d trees* [3], which try to get a partition as balanced as possible of the data space.

Median  $K$ -d trees and hybrid median  $K$ -d trees also aim to build a more balanced tree. In median  $K$ -d trees the rule is to choose as discriminant at each node the coordinate that would presumably divide the forthcoming elements as evenly as possible into the two subtrees of the node. While this can be easily accomplished if we have the collection of  $n$  items beforehand, median  $K$ -d trees achieve a similar outcome using a heuristic based on the usual assumption that the keys from which the tree is built are drawn uniformly at random in  $[0, 1]^K$ . Besides, hybrid median  $K$ -d trees combine the heuristics of standard and median  $K$ -d trees: at every node the coordinate used to discriminate is chosen using the median  $K$ -d tree heuristic but, in a cyclic way as in standard  $K$ -d trees.

We use here the *internal path length* (IPL)<sup>4</sup> [7] of median  $K$ -d trees and hybrid median  $K$ -d trees as a measure of their degree of balance and of the cost of building the tree and of exact (successful) searches. As general purpose data structures,  $K$ -d trees provide efficient (not necessarily optimal and only on expectation) support for dynamic insertions, exact searches and several associative queries. In particular, we focus here on *random partial match queries* (random PM queries), first because of their own intrinsic interest and second because their analysis is a fundamental block in the analysis of other kind of associative queries such as orthogonal range and nearest neighbour queries.

<sup>3</sup> We have omitted on purpose what to do with elements  $\mathbf{v}$  such that  $x_j = v_j$ ; several alternatives exist to cope with such situation, but in the random model which we will use for the analysis such event does not occur and hence the strategy used to cope with such situation becomes unimportant.

<sup>4</sup> The internal path length of a binary search tree is the sum, over all its internal nodes, of the paths from the root to every node of the tree.

A random PM query is a pair  $\langle \mathbf{q}, \mathbf{u} \rangle$ , where  $\mathbf{q} = (q_0, \dots, q_{K-1})$  is a  $K$ -dimensional point independently drawn from the same continuous distribution as the data points, and  $\mathbf{u} = (u_0, \dots, u_{K-1})$  is the *pattern* of the query; each  $u_i = S$  (the  $i$ -th attribute of the query is *specified*) or  $u_i = *$  (the  $i$ -th attribute is *unspecified*). The goal of the PM search is to report all data points  $\mathbf{x} = (x_0, \dots, x_{K-1})$  in the tree such that  $x_i = q_i$  whenever  $u_i = S$  where  $s$  is the number of specified coordinates; the interesting cases arise when  $0 < s < K$ .

Our main tool for the analysis of the expected IPL and the expected cost of random PMs is the continuous master theorem (CMT, for short) [13] and some “extensions” developed here to cope with systems of divide-and-conquer recurrences. In particular, we give the main order term of the expected IPL of median  $K$ -d trees and hybrid median  $K$ -d trees: in both cases it is of the form  $\sim c_K n \log_2 n$  for a constant  $c_K$  depending on  $K$  and on the variant of  $K$ -d tree considered (Theorems 1 and 3); median  $K$ -d trees and hybrid median  $K$ -d trees perform better than other variants, for all  $K \geq 2$ , since  $c_K < 2$  —while  $c_K = 2$  for all  $K$  in standard, relaxed and squarish  $K$ -d trees. Moreover, in median  $K$ -d trees and hybrid median  $K$ -d trees  $c_K \rightarrow 1$  as  $K \rightarrow \infty$ , which is optimal for data structures built using key comparisons. We also show that the expected cost of random PM searches will be always  $\Theta(n^\alpha)$  for an exponent  $\alpha$  which depends on the variant of  $K$ -d trees, the dimension  $K$  and the number  $s$  of coordinates which are specified in the PM query. We give the equations satisfied by the exponent  $\alpha$  in each case (Theorems 2 and 4). Although in general these equations are not analytically solvable, it is possible to provide accurate numerical approximations. In the case of median  $K$ -d trees, the expected cost of PM queries lies somewhere between that of standard  $K$ -d trees and that of relaxed  $K$ -d trees, and  $\alpha \rightarrow \log_2(2 - s/K)$  as  $K \rightarrow \infty$ . For hybrid median  $K$ -d trees the expected PM cost outperforms that of relaxed and standard  $K$ -d trees for all  $K \geq 2$ , and we conjecture that  $\alpha \rightarrow 1 - s/K$  as  $K \rightarrow \infty$ , which is optimal. Table 1 summarizes our results comparing them to other variants of  $K$ -d trees.

**Table 1.** An abridged comparison of median  $K$ -d trees and hybrid median  $K$ -d trees with other families of  $K$ -d trees, giving the coefficient of  $n \log n$  for IPL and the exponent  $\alpha$  for PM where \* indicates conjectured.

Family	IPL		Partial match $s = 1, s = K/2,$	
	$K = 2$	$K \rightarrow \infty$	$K = 2$	$K \rightarrow \infty$
Standard $K$ -d trees [1,6]	2	2	0.56155	0.56155
Relaxed $K$ -d trees [4,10]	2	2	0.618	0.618
Squarish $K$ -d trees [3]	2	2	0.5	0.5
Median $K$ -d trees [this paper]	1.66	$\rightarrow 1.443$	0.602	$\rightarrow 0.585$
Hybrid median $K$ -d trees [this paper]	1.814	$\rightarrow 1.443$	0.546	$\rightarrow 0.5^*$

This paper is organized as follows. In Section 2 we give the definition of random median  $K$ -d trees as well as the previous known results on them and we present the analysis of their expected IPL (Subsec. 2.1) and the expected cost of random PMs (Subsec. 2.2). In Section 3 we proceed as in the preceding section, now with the analysis of random hybrid median  $K$ -d trees. Finally, in Section 4 we give our conclusions and guidelines for future work.

## 2 Median $K$ -d Trees

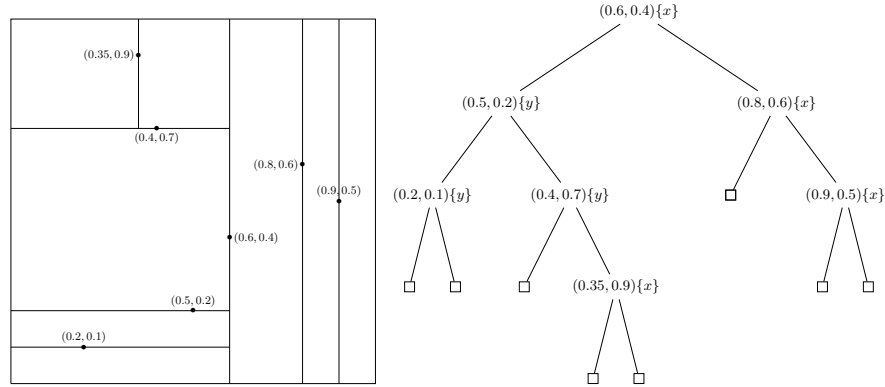
Median  $K$ -d trees were introduced by Pons [12] and they are a simple variant of standard  $K$ -d trees: the only difference lies in the way to choose the dimension used to discriminate at each node.

As happens in plane binary search trees, in  $K$ -d trees the insertion of an item creates a new node that replaces a leaf of the current tree. It is worth noting that every leaf of a  $K$ -d tree corresponds to a region of the space from which the elements are drawn and hence the whole tree induces a partition of the space  $[0, 1]^K$  in our case. The region delimited by the leaf that a new node replaces at the moment of its insertion into the tree is known as its *bounding box*.

In median  $K$ -d trees, when a new data point  $\mathbf{x} = (x_0, \dots, x_{K-1})$  is inserted in the bounding box  $R = [\ell_0, u_0] \times \dots \times [\ell_{K-1}, u_{K-1}]$  the discriminant  $j$  is chosen as follows,

$$j = \arg \min_{0 \leq i < K} \left\{ \left| \frac{x_i - \ell_i}{u_i - \ell_i} - \frac{1}{2} \right| \right\}.$$

An example of median  $K$ -d tree together with its induced partition of the space is shown in Fig. 1.



**Fig. 1.** Example of a median  $K$ -d tree built from 2-dimensional points.

In the analysis of the expected IPL and the expected cost of random PM in a median  $K$ -d tree of size  $n$ , we will assume, as usual in the literature, that the

tree is randomly built. That is, that the  $n$  points are random and independently drawn from  $[0, 1]^K$ , with each coordinate  $x_i$  of a data point  $\mathbf{x}$  independently and uniformly drawn from  $[0, 1]$ .

In [12] it is shown that (i) the expected IPL of random median  $K$ -d trees is  $I_n \sim c_K n \log_2 n$  for a constant  $c_K$  depending on  $K$ ; it is also stated there without formal proof that  $c_K \rightarrow 1$  as  $K \rightarrow \infty$ ; and (ii) that, for  $K = 2$  and  $K = 3$ , the expected cost of a random PM is  $\Theta(n^\alpha)$  with  $\alpha(1, 2) \approx 0.60196\dots$ ,  $\alpha(2, 3) \approx 0.74387\dots$  and  $\alpha(1, 3) \approx 0.42756\dots$ .

Here, using the CMT, we obtain the same results for the expected IPL and extend the analysis of the expected cost of random PM to any value of  $K$  and  $s$  proving also that  $\alpha \rightarrow \log_2(2 - s/K)$  as  $K$  grows (and  $s/K$  remains constant).

In order to proceed with the analysis, we need to compute the probability that the left subtree of a random median  $K$ -d tree is of size  $j$ , given that the tree is of size  $n$ . This is crucial in order to set up the recurrences for the expected IPL and the expected cost of partial matches in the next subsections, and it enables the systematic application of the CMT (see [13] or Appendix A of [5]) to solve the recurrences, instead of the ad-hoc arguments given in [12].

Let  $\mathbf{x} = (x_0, x_1, \dots, x_{K-1})$  be the key stored at the root of a median  $K$ -d tree  $T$  that contains  $n$  data points. We can define the *rank vector* of  $\mathbf{x}$  as  $\mathbf{r} = \mathbf{r}(\mathbf{x}, T) = (r_0, r_1, \dots, r_{K-1})$  where  $r_i$  is the number of data points in  $T$  with  $i$ -th coordinate smaller or equal to  $x_i$ . If the root of  $T$  discriminates with respect to the  $i$ -th coordinate then —because we assume that the tree is randomly built— the size of the left subtree  $L$  of  $T$  will be  $r_i - 1$  and the size of the right subtree will be  $n - r_i$ . In an idealization of median  $K$ -d trees the chosen discriminating coordinate will be  $i$  if  $r_i$  is the closest rank to  $\lfloor (n+1)/2 \rfloor$  —ties are resolved in favor of the coordinate with smallest index. It follows that

$$\mathbb{P}\{|L| = j \mid |T| = n\} = \mathbb{P}\{Z_{n,K} = j + 1\}, \quad 0 \leq j < |T|,$$

where  $Z_{n,K}$  denotes the closest integer to  $\lfloor (n+1)/2 \rfloor$  (equivalently the closest integer to  $\lceil n/2 \rceil$ ) in a set of  $K$  given integers independently and uniformly drawn from  $\{1, \dots, n\}$ .

For example, let  $K = 2$  and  $Z := Z_{n,2}$ . Then we have

$$\mathbb{P}\{Z = j\} = \begin{cases} \frac{4j-1}{n^2} & \text{if } j \leq \lfloor \frac{n}{2} \rfloor, \\ \frac{4(n-j)+1}{n^2} & \text{if } j > \lfloor \frac{n}{2} \rfloor. \end{cases} \quad (1)$$

To see why, suppose that  $n = 2\lambda + 1$  and  $j \leq \lambda = \lfloor n/2 \rfloor$ . Then  $Z = j$  will occur if (1) both ranks are equal to  $j$ , this happens with probability  $1/n^2$  or (2) one rank is  $j$  the other is  $< j$  or  $\geq n + 1 - j$ , which will happen with probability  $2 \cdot (1/n) \cdot (j - 1 + j)/n = (4j - 2)/n^2$ . Hence the probability of  $Z = j$  when  $j \leq \lambda$  is  $(4j - 1)/n^2$ . The case for  $j > \lambda + 1$  is similar except that ties in the distance to the center are resolved in favor of the smallest rank: thus if  $j > \lambda + 1$  then  $n - j + 1$  will be at the same distance to the center but smaller than  $j$  hence  $Z = j$  requires one rank to be  $j$  and the other be smaller than  $n + 1 - j$ . Thus, the probability that  $Z = j$  when  $j > \lambda + 1$  is

$1/n^2 + 2 \cdot 2(n-j)/n^2 = (4(n-j) + 1)/n^2$ . On the other hand, if  $j = \lambda + 1$  then we will have  $Z = j$  no matter what the other rank is; we have that the probability of  $Z = \lambda + 1$  is  $1/n^2 + 2(n-1)/n^2 = (2n-1)/n^2 = (4\lambda+1)/n^2 = (4(n-j)+1)/n^2$ . Therefore, we can write that the probability of  $Z = j$  when  $j \geq \lambda + 1 > \lfloor n/2 \rfloor$  is  $(4(n-j) + 1)/n^2$ . For even  $n$ , when  $n = 2\lambda$ , the arguments are identical and Eq. (1) holds too.

For the general case, we can reason in an analogous way, assuming that  $\ell \geq 1$  of the  $K$  ranks are  $j$  and  $K - \ell$  are either smaller than  $j$  or greater than  $n - j$ . If  $j \leq \lfloor n/2 \rfloor$  then

$$\mathbb{P}\{Z = j\} = \frac{1}{n^K} \cdot [(2j)^K - (2j-1)^K],$$

and if  $j > \lfloor n/2 \rfloor$  then the analysis is analogous but we need a small correction as we cannot allow any coordinate to be  $n + 1 - j$ , hence in that case

$$\mathbb{P}\{Z = j\} = \frac{1}{n^K} \cdot [(2(n-j) + 1)^K - (2(n-j))^K].$$

## 2.1 Internal Path Length

Let us start writing down the recurrence for the expected IPL  $I_n$  of a random median  $K$ -d tree  $T$  of size  $n$ , for  $n > 0$ . For that, we condition on the size of the left subtree  $L$ , thus

$$\begin{aligned} I_n &= n - 1 + \sum_{j=0}^{n-1} \pi_{n,j} (I_j + I_{n-1-j}) = n - 1 + \sum_{j=0}^{n-1} \pi_{n,j} I_j + \sum_{j=0}^{n-1} \pi_{n,n-1-j} I_j \\ &= n - 1 + \sum_{j=0}^{n-1} (\pi_{n,j} + \pi_{n,n-1-j}) I_j, \end{aligned} \quad (2)$$

with  $\pi_{n,j} = \mathbb{P}\{|L| = j \mid |T| = n\} = \mathbb{P}\{Z_{n,K} = j + 1\}$  and  $I_0 = 0$ . Indeed, the IPL of  $T$  is the sum of the IPL of its subtrees  $L$  and  $R$ , and we add  $+1$  for every internal node other than the root. In order to apply the continuous master theorem we identify  $\omega_{n,j} = \pi_{n,j} + \pi_{n,n-1-j}$  as the weights sequence in the divide-and-conquer recurrence. Substituting  $j$  by  $z \cdot n$ , multiplying by  $n$  and taking the limit when  $n \rightarrow \infty$  we get the *shape function*

$$\omega_K(z) = \lim_{n \rightarrow \infty} n \cdot \omega_{n,z \cdot n} = \begin{cases} 2K(2z)^{K-1} = K2^K z^{K-1}, & \text{if } z \leq 1/2, \\ 2K(2(1-z))^{K-1} = K2^K (1-z)^{K-1}, & \text{if } z \geq 1/2. \end{cases}$$

When  $n \rightarrow \infty$ , the shape function derived for the idealization using ranks is the actual shape function for median  $K$ -d trees, where we would have had to compute the probability that, given a random set of  $K$  points  $X_0, \dots, X_{K-1}$  independently and uniformly drawn from  $[0, 1]$ , we have  $Z'_{n,K} = j$  with

$$Z'_{n,K} = \#\{X_i \mid X_i < X_\ell\},$$

where  $\ell = \arg \min_{0 \leq i < K} \{|X_i - 1/2|\}$ .

Once we have the shape function for the divide-and-conquer recurrence, we can get the const-entropies for all  $K \geq 1$ :

$$\mathcal{H}_K = 1 - \int_0^1 z \omega_K(z) dz = 0.$$

As they all are zero, we need to compute the log-entropies:

$$\mathcal{H}'_K = - \int_0^1 z \ln(z) \omega_K(z) dz. \quad (3)$$

No easy closed form for  $\mathcal{H}'_K$  is available; but we can compute any value of  $\mathcal{H}'_K$  and thus of the expected IPL (see Table 2).

**Theorem 1 (Pons, 2010).** *The expected IPL of random median  $K$ -d tree of size  $n$  is*

$$I_n = c_K n \ln n + o(n \log n)$$

where

$$c_K^{-1} = \mathcal{H}'_K = -K 2^K \left[ A_K + \sum_{0 \leq i < K} \binom{K-1}{i} (-1)^i B_{i+1} \right],$$

with  $B_j = -(A_j + 1/(j+1)^2)$  and

$$A_j = \int_0^{1/2} z^j \ln z dz = -\frac{1 + (j+1) \ln 2}{2^{j+1}(j+1)^2},$$

The IPL gives a measure of the cost of building the  $K$ -d tree in the first place, but also of the cost of exact successful searches. Indeed,  $\frac{I_n}{n} = c_K \cdot \ln n + o(\log n)$  is the expected depth of a random node. We can use the definition of  $\mathcal{H}'_K$  to show that  $\mathcal{H}'_K < \mathcal{H}'_{K+1}$  and thus the coefficients  $c_K = (\mathcal{H}'_K)^{-1}$  are monotonically decreasing with  $K$ . It is also easy to prove that  $c_K \rightarrow 1/\ln 2$  which implies that median  $K$ -d trees tend to get perfectly balanced, as  $K \rightarrow \infty$  (see Fig. 3 on page 12). Indeed, from the definition (3) of  $\mathcal{H}'_K$ , if we let  $K \rightarrow \infty$  the shape function under the integral sign degenerates to a Dirac's delta distribution at  $z = 1/2$  and thus

$$\mathcal{H}'_K \rightarrow - \int_0^1 \ln z \delta_{1/2}(z) dz = -\ln(1/2) = \ln 2.$$

## 2.2 Random Partial Match

Consider a random partial match with  $s$  specified coordinates,  $0 < s < K$ . Because of the symmetries of the problem all the coordinates are equivalent with respect to the query pattern and thus we can assume without loss of generality that the query is of the form  $\mathbf{q} = (q_0, \dots, q_{s-1}, *, \dots, *)$  with  $q_i$  a uniformly

**Table 2.** Coefficient of the first order term in the expected IPL of random median  $K$ -d trees.

$K$	$\mathcal{H}'_K$	$\mathbb{E}\{I_n\}/(n \ln n) \sim c_K = 1/\mathcal{H}'_K$
1	1/2	2
2	$5/6 - 1/3 \ln 2 \approx 0.6023$	1.660
3	$4/3 - \ln 2 \approx 0.6402$	1.562
4	$131/60 - 11/5 \ln 2 \approx 0.6584$	1.519
...	...	...
$\infty$	$\ln 2 \approx 0.6931$	$1/\ln 2 \approx 1.443$

drawn real number in  $[0, 1]$ . Then the recurrence for the expected cost  $P_n := P_n^{(K,s)}$  of the PM is

$$\begin{aligned}
P_n &= 1 + \frac{s}{K} \sum_{j=0}^{n-1} \pi_{n,j} \left( \frac{j+1}{n+1} P_j + \frac{n-j}{n+1} P_{n-1-j} \right) + \frac{K-s}{K} \sum_{j=0}^{n-1} \pi_{n,j} (P_j + P_{n-1-j}) \\
&= 1 + \frac{s}{K} \sum_{j=0}^{n-1} (\pi_{n,j} + \pi_{n,n-1-j}) \frac{j+1}{n+1} P_j + \frac{K-s}{K} \sum_{j=0}^{n-1} (\pi_{n,j} + \pi_{n,n-1-j}) P_j.
\end{aligned} \tag{4}$$

To derive the recurrence above, we condition on the size of the left subtree, and consider two possibilities: with probability  $s/K$  the discriminating coordinate of the root is specified, and we have to continue recursively in the left or the right subtree with probability proportional to their number of leaves of each subtree. On the other hand, with probability  $(K-s)/K$  the discriminating coordinate of the root is not specified and the PM must continue in both subtrees. We have thus that the shape function is

$$\omega_K(z) = \begin{cases} K 2^K z^{K-1} (\rho z + 1 - \rho), & \text{if } z \leq 1/2, \\ K 2^K (1-z)^{K-1} (\rho z + 1 - \rho), & \text{if } z \geq 1/2, \end{cases}$$

with  $\rho := s/K \in (0, 1)$ . Then the const-entropy is

$$\mathcal{H}_K = 1 - \int_0^1 \omega_K(z) dz = \rho - 1,$$

which is always negative, since  $\rho < 1$ . In this situation the CMT tells us that the expected PM cost will be  $P_n = \Theta(n^\alpha)$ , where  $\alpha$  is the unique root in  $[0, 1]$  of the equation

$$\int_0^1 z^\alpha \omega_K(z) dz - 1 = 0,$$

**Theorem 2.** *The expected cost of a random partial match with  $s$  specified coordinates out of  $K$ ,  $0 < s < K$ , in a random median  $K$ -d tree of size  $n$  is*

$P_n = \Theta(n^\alpha)$ , where  $\alpha \in [0, 1]$  is the unique real solution of

$$2^{-\alpha} \left( \frac{K(1-\rho)}{K+\alpha} + \frac{K\rho}{2(K+\alpha+1)} \right) + K2^K \left\{ \rho B(1/2; K+1, \alpha+1) + (1-\rho) B(1/2; K, \alpha+1) \right\} = 1, \quad (5)$$

with  $B(z; a, b) = \int_0^z t^{a-1}(1-t)^{b-1} dt$  denoting the incomplete Beta function [11, Ch. 8] and  $\rho = s/K$ .

While we cannot give a closed form for  $\alpha$  in terms of  $K$  and  $\rho$ , Equation (5) can be used to compute numerical approximations with a high degree of accuracy.

We can also find the value of  $\alpha$  as  $K$  grows and  $\rho = s/K$  remains constant. For very large  $K$ , known asymptotic expansions of the incomplete Beta function (see for instance [9] or [11, Ch. 8, pp. 183–184]) yield that  $\alpha$  must satisfy

$$2^{-\alpha} \left( 1 - \rho + \frac{\rho}{2} \right) + K2^K \left( \frac{1}{2} \right)^\alpha \frac{1}{K2^K} (\rho/2 + 1 - \rho) = 2^{-\alpha} (2 - \rho) = 1,$$

and hence  $\alpha = \log_2(2 - \rho)$ . In it is interesting to note that it coincides with the exponent of the expected cost of random PM in relaxed  $K$ -d tries [10].

Fig. 2 plots the excess  $\vartheta(x) := \alpha(x) - (1 - x)$  in the exponent of the cost of random PM of median  $K$ -d trees for various values of  $K$  (and  $x \equiv s/K$ ), and, for comparison, we also plot the excess  $\vartheta(x)$  for relaxed  $K$ -d trees [4, 10], standard  $K$ -d trees [6] and the limit curve  $\log_2(2 - x) - 1 + x$  that corresponds to the excess in the exponent for relaxed  $K$ -d tries [10].

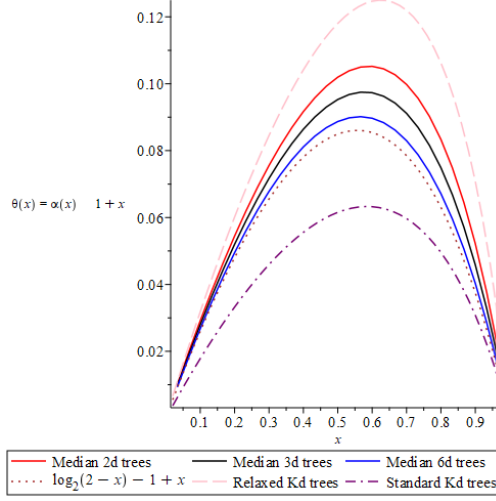
### 3 Hybrid Median $K$ -d Trees

*Hybrid  $K$ -d trees*, also introduced in [12], combine two different rules to choose discriminants. In particular, the hybridization of median  $K$ -d trees with standard  $K$ -d trees are the so called hybrid median  $K$ -d trees, where, for an arbitrary dimension  $K \geq 2$ , the rule to assign the discriminants is the following:

1. Nodes at levels  $\ell \equiv 0 \pmod{K}$  discriminate with respect to the median rule applied to all  $K$  coordinates
2. Nodes at levels  $\ell \equiv j \pmod{K}$ ,  $0 < j < K$ , discriminate with respect the median rule applied to all the coordinates not used as discriminant by any of its  $j - 1$  immediate ascendants.

The above implies that, in such a tree, in any path from the root to a leaf, looking at the discriminants of the nodes along the path we will find a sequence of permutations of order  $K$  (except for the last part of the path, which will eventually contain only  $j < K$  distinct discriminants).

The analysis of the IPL and random partial match in hybrid median  $K$ -d trees now becomes more complicated as it requires considering a system of divide-and-conquer recurrences instead of a single divide-and-conquer recurrence as we had when analyzing median  $K$ -d trees.



**Fig. 2.** The excess  $\vartheta(x) = \alpha(x) - 1 - x$  for various median  $K$ -d trees and other  $K$ -d trees.

### 3.1 Internal Path Length

Let us consider first the IPL of an hybrid median  $K$ -d tree. Let  $I_n^{(\ell)}$  denote the expected IPL of an hybrid median  $K$ -d tree of size  $n$  where there are only  $\ell$  available choices for the discriminant at the root (because the other  $K - \ell$  discriminants have been already used for the immediate ancestors), then the probability that the left subtree is of size  $j$  is given by  $\pi_{n,j}^{(\ell)} = \mathbb{P}\{Z_{n,\ell} = j + 1\}$  and if  $\ell > 1$  we have

$$I_n^{(\ell)} = n - 1 + \sum_{j=0}^{n-1} \pi_{n,j}^{(\ell)} \left( I_j^{(\ell-1)} + I_{n-1-j}^{(\ell-1)} \right), \quad 1 < \ell \leq K \text{ and } n > 0,$$

and

$$I_n^{(1)} = n - 1 + \sum_{j=0}^{n-1} \pi_{n,j}^{(1)} \left( I_j^{(K)} + I_{n-1-j}^{(K)} \right), \quad n > 0.$$

Define now the sequences of vectors  $\mathbf{F}_n = (I_n^{(K)}, \dots, I_n^{(1)})^T$  and  $\mathbf{t}_n = (n - 1, \dots, n - 1)^T$ , and the sequence of weight matrices  $\mathbf{\Omega}_{n,k} = (\omega_{n,k}^{(i,j)})_{K \times K}$ , where  $\omega_{n,k}^{(i,i+1)} = \pi_{n,k}^{(K+1-i)} + \pi_{n,n-1-k}^{(K+1-i)}$  if  $i < K$ ,  $\omega_{n,k}^{(K,1)} = \pi_{n,k}^{(1)} + \pi_{n,n-1-k}^{(1)}$  and all other  $\omega_{n,k}^{(i,j)} = 0$ . Then we can compactly express the system for the IPL as

$$\mathbf{F}_n = \mathbf{t}_n + \sum_{0 \leq k < n} \mathbf{\Omega}_{n,k} \cdot \mathbf{F}_k.$$

Let us suppose that we substitute in the recurrences above each  $F_k^{(i)} \equiv I_k^{(i)}$  by its corresponding “row” in the system. This substitution can be expressed in terms of the following operation between weight sequences  $\{\omega_{n,k}\}$  and  $\{\omega'_{n,k}\}$ , giving a new sequence  $\{\omega''_{n,k}\}$  defined by

$$\omega''_{n,k} = (\omega \otimes \omega')_{n,k} := \sum_{k < j < n} \omega_{n,j} \cdot \omega'_{j,k}.$$

The operation can be naturally extended to sequences of square  $d \times d$  matrices ( $d = K$  in our instance). The  $(i, j)$  component of each matrix in the sequence  $\{\tilde{\Omega}_{n,k}\} := \{(\Omega \otimes \hat{\Omega})_{n,k}\} = \{\Omega_{n,k}\} \otimes \{\hat{\Omega}_{n,k}\}$  is given by

$$\tilde{\Omega}_{n,k}^{(i,j)} = (\Omega \otimes \hat{\Omega})_{n,k}^{(i,j)} = \sum_{\ell} \left( \omega^{(i,\ell)} \otimes \hat{\omega}^{(\ell,j)} \right)_{n,k}.$$

Then we can write one substitution step as

$$\mathbf{F}_n = \mathbf{t}_n + \sum_{0 \leq k < n} \Omega_{n,k} \cdot \mathbf{t}_k + \sum_{0 \leq k < n} (\Omega \otimes \Omega)_{n,k} \cdot \mathbf{F}_k$$

The substitution process can be iterated repeatedly:

$$\begin{aligned} \mathbf{F}_n = \mathbf{t}_n + \sum_{0 \leq k < n} \Omega_{n,k} \cdot \mathbf{t}_k + \sum_{0 \leq k < n} \Omega_{n,k}^{[2]} \cdot \mathbf{t}_k + \cdots + \sum_{0 \leq k < n} \Omega_{n,k}^{[\ell-1]} \cdot \mathbf{t}_k \\ + \sum_{0 \leq k < n} \Omega_{n,k}^{[\ell]} \cdot \mathbf{F}_k, \end{aligned}$$

where  $\Omega^{[1]} \equiv \Omega$  and  $\Omega^{[\ell]} = \Omega \otimes \Omega^{[\ell-1]}$ , for  $\ell > 1$ . This new operation  $\otimes$ —let us call it *substitution product*—of weight sequences is associative and commutative, and distributes respect to the sum. Its extension to matrices is associative but not commutative, exactly as ordinary matrix products. In the case of the IPL of hybrid  $K$ -d trees it turns out that the matrix  $\Omega_{n,k}^{[K]}$  is diagonal. This is a very lucky circumstance since then we obtain a set of  $K$  independent divide-and-conquer recurrences, and each one can be readily solved using the CMT. To that end, we would only need to compute the weight matrix  $\Omega_{n,k}^{[K]}$  and the new toll function

$$\hat{\mathbf{t}}_n = \mathbf{t}_n + \sum_{0 \leq k < n} \left( \Omega_{n,k} + \Omega_{n,k}^{[2]} + \cdots + \Omega_{n,k}^{[K-1]} \right) \cdot \mathbf{t}_k.$$

Rather than computing  $\Omega_{n,k}^{[\ell]}$  for all  $\ell > 1$ , the special structure of the problem can be further exploited to obtain our final result (Theorem 3 below, whose proof is given in Appendix B of [5]). In particular, to prove the theorem we introduce the *shape matrix*  $\Omega(z)$  in which the  $(i, j)$  entry is the shape function for the sequence  $\{\omega_{n,k}^{(i,j)}\}$  and the matrices

$$\Phi_{\ell}(x) = \left( \int_0^1 (\Omega^{[\ell]}(z))^{(i,j)} z^x dz \right)_{K \times K}, \quad \Phi'_{\ell}(x) = \left( - \int_0^1 (\Omega^{[\ell]}(z))^{(i,j)} z^x \ln z dz \right)_{K \times K}$$

which are the  $K$ -dimensional analogous of the const- and log-entropies of the CMT. Properties of  $\otimes$  (such as those proven in Appendix C of [5]) are used to simplify the calculation and show that  $\mathbf{F}_n \sim (\boldsymbol{\Phi}'_K(1))^{-1} \hat{\mathbf{t}}_n \ln n + o(n \log n)$ , where  $\hat{\mathbf{t}}_n = (Kn, Kn, \dots, Kn)^T + o(\mathbf{1})$ . We also show that  $\boldsymbol{\Phi}'_K(1)$  is a diagonal matrix where all non-null entries are equal to  $\mathcal{H}'_1 + \dots + \mathcal{H}'_K$ , with  $\mathcal{H}'_i$  the log-entropy for the expected IPL in median  $i$ -dimensional trees.

**Theorem 3.** *The expected IPL of a random hybrid median  $K$ -d tree of size  $n$  is*

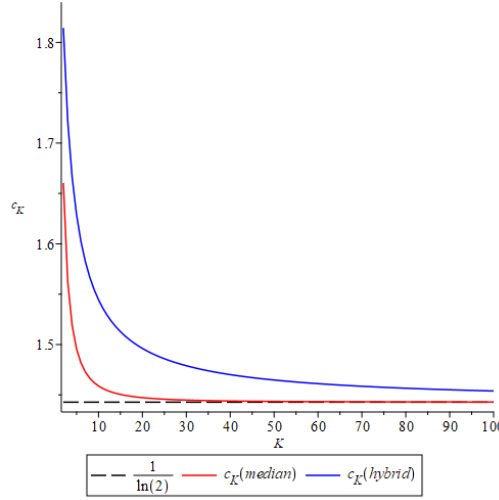
$$I_n = c_K^{[hm]} n \ln n + o(n \log n)$$

where

$$c_K^{[hm]} = \frac{K}{\mathcal{H}'_1 + \dots + \mathcal{H}'_K},$$

and the values of  $\mathcal{H}'_i$  are those given in Theorem 1.

To conclude, let us observe that for all  $K$ ,  $c_K^{[hm]} \geq c_K^{[med]} = \frac{1}{\mathcal{H}'_K}$  and also that  $c_K^{[hm]} \rightarrow 1/\ln 2$ , albeit the convergence speed is slower than for median  $K$ -d trees (as can be seen in Fig. 3).



**Fig. 3.** The coefficient of  $n \ln n$  in the average IPL of median  $K$ -d trees (red) and hybrid median  $K$ -d trees (blue).

### 3.2 Random Partial Match

Let  $P_n^{(i,\ell)}$  denote the expected cost of a random PM in a hybrid median  $K$ -d tree of size  $n$  in which there are only  $i \geq 1$  coordinates to choose as discriminants —

the remaining  $K - i$  have been used in the immediate ancestors— and  $0 \leq \ell \leq i$  of them are specified in the query. We are interested in  $P_n^{(K,s)}$  with  $0 < s < K$ .

Suppose  $i > \ell \geq 1$ . With probability  $\ell/i$  the discriminant coordinate — chosen by the median rule among  $i$  choices— is specified and thus we will either continue in the left subtree of size  $j$  or the right subtree of size  $n - 1 - j$  with probability  $\pi_{n,j}^{(i)} \frac{j+1}{n+1}$  or  $\pi_{n,n-1-j}^{(i)}$ , respectively, but now in the next level we will be paying the expected cost of a random PM with only  $i - 1$  available coordinates of which only  $\ell - 1$  are specified. On the other hand, with probability  $(i - \ell)/i$  the discriminant won't be specified and the recursion will continue in both subtrees with only  $i - 1$  available coordinates to chose from to discriminate but still  $\ell$  coordinates specified. If  $i = \ell > 0$  the reasoning above applies with only branching to either the left or the right subtrees; and if  $i > \ell = 0$  then we will continue in both subtrees as no specified coordinate is among those that can be used as discriminants. Hence, if  $i > 1$  and  $0 \leq \ell \leq i$  we have

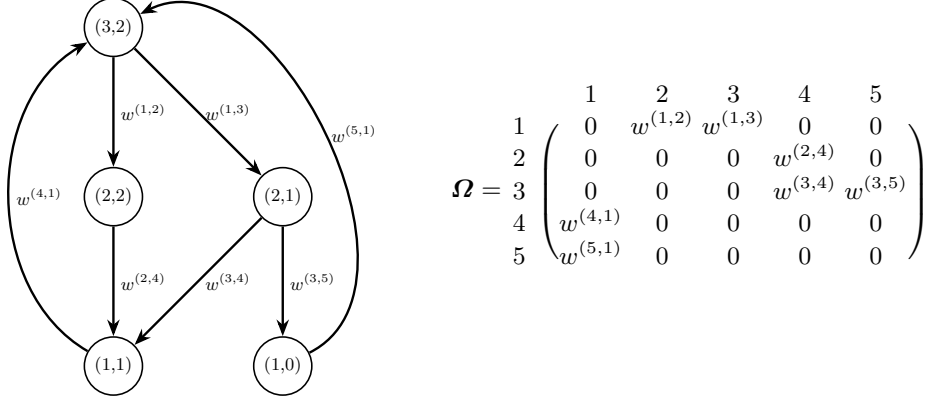
$$\begin{aligned} P_n^{(i,\ell)} &= 1 + \frac{\ell}{i} \sum_{j=0}^{n-1} \left( \pi_{n,j}^{(i)} + \pi_{n,n-1-j}^{(i)} \right) \frac{j+1}{n+1} P_j^{(i-1,\ell-1)} \\ &\quad + \frac{i-\ell}{i} \sum_{j=0}^{n-1} \left( \pi_{n,j}^{(i)} + \pi_{n,n-1-j}^{(i)} \right) P_j^{(i-1,\ell)} \end{aligned}$$

The special cases are thus: (1) when  $i = 1$  and  $\ell = 1$ , then the recursion follows in the appropriate subtree but all the  $K$  discriminants become available in the next level; and (2) when  $i = 1$  and  $\ell = 0$ , then the recursion follows in both subtrees but with all  $K$  coordinates again usable to discriminate. That is,

$$\begin{aligned} P_n^{(1,0)} &= 1 + \sum_{j=0}^{n-1} \left( \pi_{n,j}^{(1)} + \pi_{n,n-1-j}^{(1)} \right) P_j^{(K,s)} \\ P_n^{(1,1)} &= 1 + \sum_{j=0}^{n-1} \left( \pi_{n,j}^{(1)} + \pi_{n,n-1-j}^{(1)} \right) \frac{j+1}{n+1} P_j^{(K,s)} \end{aligned}$$

The resulting call graph is more complicated than the one of IPL, and the system of D&C recurrences will involve  $d = (K - s + 1)(s + 1) - 1$  “algorithms” with costs  $P_n^{(i,\ell)}$ , see for example Fig. 4 for the case  $K = 3$  and  $s = 2$ .

Once we have set up the system of divide-and-conquer recurrences we can construct a shape matrix  $\Omega(z)$  in which the entry  $(u, v)$  is the shape function  $\omega^{(u,v)}(z)$  corresponding to weight sequence  $\omega_{n,k}^{(u,v)}$ ; vertices  $u$  and  $v$  correspond to partial match algorithms with parameters  $(i, \ell)$  and  $(i', \ell')$ . Many entries will be null as algorithm  $u$  (or  $(i, \ell)$ ) does not call algorithm  $v$  (or  $(i', \ell')$ ). We can think of this shape matrix as the adjacency matrix for the call digraph in which each edge  $(u, v)$  is labelled by  $\omega^{(u,v)}(z)$ . Likewise we can define the matrix  $\Phi(x)$  in which the entries are the definite integrals  $\int_0^1 \omega^{(u,v)}(z) z^x dz$ . Then we can find the expected cost  $P_n^{(K,s)}$  thanks to the following result.



**Fig. 4.** Call graph for the system of D&C recurrences of the PM costs in hybrid median  $K$ -d trees for  $K = 3$  and  $s = 2$ .

**Theorem 4.** *The expected cost of a random partial match with  $s$  specified coordinates out of  $K$ ,  $0 < s < K$ , in a random hybrid median  $K$ -d tree of size  $n$  is  $P_n^{(K,s)} = \Theta(n^\alpha)$ , where  $\alpha \in [0, 1]$  is the unique real solution of  $\det(\mathbf{I} - \Phi(x)) = 0$ , where  $\Phi(x) = \int_0^1 \Omega(z) z^x dz$  and  $\Omega(z)$  is the shape matrix corresponding to the system of  $d$  divide-and-conquer recurrences, with  $d = (K - s + 1)(s + 1) - 1$ .*

The proof of this result can be found in Appendix D of [5]. It is based in the properties of iterated substitution matrices  $\Omega^{[K]}(z)$  (and  $\Phi_K(x)$ ), and those of the determinant of  $\Phi(x) - \mathbf{I}$  once we see it as the (weighted) adjacency matrix of the call graph in which we add self-loops to every vertex of the call graph.

We report the values of  $\alpha$  for  $K \leq 6$  in Table 3. Next to each entry we give inside parentheses the corresponding values of  $\alpha$  for standard  $K$ -d trees. All values have been rounded to three significant figures. These values suggest that random PM in hybrid median  $K$ -d trees perform better on average than in standard  $K$ -d trees. Namely, we conjecture that  $\alpha^{[\text{hyb}]}(s, K) < \alpha^{[\text{std}]}(s, K)$  for all  $s$  and  $K$ . Moreover, we conjecture that as  $K \rightarrow \infty$ ,  $\alpha^{[\text{hyb}]}(s, K) \rightarrow 1 - s/K$ , which is optimal. One argument in favor of this conjecture is that hybrid median  $K$ -d trees get increasingly balanced as  $K$  grows, but the hybridization guarantees that we cycle over all  $K$  coordinates as we follow paths down the tree—any path from a node at level  $r \cdot K$  to a node at level  $(r + 1) \cdot K - 1$  has used all coordinates as discriminants. Hence partial match in hybrid median  $K$ -d trees should behave as in standard  $K$ -d tries [6], for which  $\alpha(s, K) = 1 - s/K$ .

## 4 Conclusions and Final Remarks

Throughout this work we have considered two variants of  $K$ -d trees: median  $K$ -d trees and hybrid median  $K$ -d trees. Both are simple and easy to implement,

**Table 3.** Values of  $\alpha$  for the expected cost of PM in hybrid median  $K$ -d trees. In parentheses, the corresponding values of  $\alpha$  for standard  $K$ -d trees.

	$s$				
$K$	1	2	3	4	5
2	0.546 (0.562)	-	-	-	-
3	0.697 (0.716)	0.368 (0.395)	-	-	-
4	0.771 (0.79)	0.53 (0.562)	0.275 (0.306)	-	-
5	0.815 (0.833)	0.624 (0.656)	0.425 (0.463)	0.218 (0.25)	-
6	0.845 (0.862)	0.685 (0.716)	0.522 (0.562)	0.354 (0.395)	0.181 (0.211)

and neither requires significant extra space. We show that both variants are more balanced than most other well known variants of  $K$ -d trees based on key comparisons, such as standard, relaxed and squarish  $K$ -d trees. This is due to the fact that their expected IPL is  $\sim c_K n \ln n$  with  $c_K < 2$  for all  $K \geq 2$ , and  $c_K \rightarrow 1/\ln 2$  as  $K \rightarrow \infty$ , while for the other mentioned variants  $c_K = 2$ . We have also shown that their expected cost for random PM is  $\Theta(n^\alpha)$ , where  $\alpha = \alpha(s, K)$ . For median  $K$ -d trees this expected cost is better than that of relaxed  $K$ -d trees but not than that for standard  $K$ -d trees. In contrast, hybrid median  $K$ -d trees outperform standard and relaxed  $K$ -d trees and we conjecture that they approach the optimal exponent —only attained by squarish  $K$ -d trees—  $\alpha = 1 - s/K$  as  $K$  gets larger. In view of these results, good choices would be hybrid median  $K$ -d trees if the efficiency of insertions and exact searches were to be prioritized —while not deviating too much from the optimal performance in partial matches— or squarish  $K$ -d trees if the priority were the efficiency of partial match, with slightly worse expected costs for insertions and exact searches.

To derive analytic results, our main tool has been the continuous master theorem —the CMT. For the analysis of median  $K$ -d trees the most challenging step was to find the probability that a random median  $K$ -d tree of size  $n$  has a left subtree of size  $j$ , but once computed an almost direct application of the CMT provides the sought answers. Hybrid median  $K$ -d trees have posed an entirely new challenge as we have had to cope with systems of divide-and-conquer recurrences that can not be solved directly using the CMT. Nevertheless, we have been able to exploit the special structure of the systems corresponding to the IPL and the random PM in hybrid median  $K$ -d trees to find the constants  $c_K$  and the equations satisfied by the exponents  $\alpha(s, K)$  by developing a limited extension of the CMT to cope with systems of recurrences.

Last but not least, our work constitutes a new example of the power of the CMT as a fundamental tool in the analysis of algorithms: without its help the analysis of median  $K$ -d trees would be a daunting task. It would have been desirable to have a full developed set of results and tools in the spirit of the CMT to cope with systems of divide-and-conquer recurrences such as those arising in the analysis of hybrid median  $K$ -d trees. Indeed, the extensions of the CMT that we have developed in this work could constitute a first step towards this goal.

## References

1. Bentley, J.L.: Multidimensional Binary Search Trees Used for Associative Searching. *Communications of the ACM* **18**(9), 509–517 (1975)
2. Cunto, W., Lau, G., Flajolet, P.: Analysis of *kdt*-trees: *kd*-trees improved by local reorganisations. In: Dehne, F., Sack, J.R., Santoro, N. (eds.) *Workshop on Algorithms and Data Structures (WADS'89)*. *Lecture Notes in Computer Science*, vol. 382, pp. 24–38. Springer-Verlag (1989)
3. Devroye, L., Jabbour, J., Zamora-Cura, C.: Squarish *k*-d trees. *SIAM Journal on Computing* **30**, 1678–1700 (2000)
4. Duch, A., Estivill-Castro, V., Martínez, C.: Randomized *K*-dimensional binary search trees. In: Chwa, K.Y., Ibarra, O.H. (eds.) *9th Annual International Symposium on Algorithms and Computation (ISAAC 98)*. *Lecture Notes in Computer Science*, vol. 1533, pp. 199–208 (1998)
5. Duch, A., Martínez, C., Pons, M., Roura, S.: The analysis of median and hybrid median *K*-dimensional trees: Two heuristically balanced variants of *K*-dimensional trees. Available from ResearchGate (2022). <https://doi.org/10.13140/RG.2.2.12891.44322>, Preprint with the full version of this extended abstract.
6. Flajolet, P., Puech, C.: Partial match retrieval of multidimensional data. *Journal of the ACM* **33**(2), 371–407 (1986)
7. Knuth, D.E.: *The Art of Computer Programming: Sorting and Searching*, vol. 3. Addison–Wesley, 2nd edn. (1998)
8. van Kreveld, M.J., Overmars, M.H.: Divided *k*-d-trees. *Algorithmica* **6**, 840–858 (1991)
9. López, J.L., Sesma, J.: Asymptotic expansion of the incomplete beta function for large values of the first parameter. *Integral Transforms and Special Functions* **8**(3), 233–236 (1999)
10. Martínez, C., Panholzer, A., Prodinger, H.: Partial match queries in relaxed multidimensional search trees. *Algorithmica* **29**(1–2), 181–204 (2001)
11. Paris, R.B.: Incomplete Gamma and related functions. In: Olver, F.W.J., Lozier, D.W., Boisvert, R.F., Clark, C.W. (eds.) *NIST Handbook of Mathematical Functions*, chap. 8. Cambridge University Press (2010)
12. Pons, M.: *Design, Analysis and Implementation of New Variants of *K*d-trees*. Master's thesis, Universitat Politècnica de Catalunya (2010)
13. Roura, S.: Improved master theorems for divide-and-conquer recurrences. *J. ACM* **48**(2), 170–205 (2001)
14. Samet, H.: *Foundations of Multidimensional and Metric Data Structures*. Morgan Kaufmann (2006)