BINARY SEARCH TREE RECURSIONS WITH HARMONIC TOLL FUNCTIONS

ALOIS PANHOLZER† AND HELMUT PRODINGER‡

Abstract. We study recursions that are traditional in the context of binary search trees and quicksort with the (non–standard) toll function H_n.

When an explicit formula can be
found there is nothing to beat it.
In the rush to abstraction and
generalization we often forget this.
Richard Askey

1. Introduction

Recursions like

$$a_n = t_n + \frac{1}{n}\sum_{k=0}^{n-1}(a_k + a_{n-1-k}), \qquad n \ge 1,\ a_0 = 1$$

with $t_n = 1$ or $t_n = n$ are very common in analysis of algorithms and data structures, especially in the context of binary search trees and quicksort, see [9]. The extra quantities t_n are usually called the *toll function*.

In the recent paper [2] however, quantities t_n that have a logarithmic behaviour, occur. (The paper [1] also studies logarithmic costs, but the results are disjoint from ours.) To be more specific, a typical recursion from [2] is

$$T_n = 1 + S_n + \frac{1}{n}\sum_{k=0}^{n-1}(T_k + T_{n-1-k}),$$

where the quantities S_n are themselves solutions of divide–and–conquer recursions, and turn out to have logarithmic behaviour.

Although such recursions might look exotic at a first glance, they really are not, and appear naturally when e. g. a (recursive) divide–and–conquer strategy has extra costs at each level of recursion which are again given by a (different) recursion. Therefore, the recursions of interest occur typically when one recursive algorithm calls another (usually simpler) recursive algorithm.

Thus, in this paper, we undertake a thorough study of such recursions, where we consider instead of $t_n = \log n$ quantities like $t_n = H_n$, with harmonic numbers $H_n = \sum_{1\le k\le n}\frac{1}{k}$, which are much more manageable and represent the discrete analogue of the logarithm.

We refrain from investigating more exotic toll functions like $\frac{n}{\sqrt{H_n}}$, since they are not likely to occur as costs of (auxiliary) algorithms, used as subroutines in recursive algorithms, and since the

Date: January 18, 20001.

class of toll functions that we consider is a reasonably large one where one can still hope for closed form solutions.

This paper can be seen both as a tuturial, showing the interested reader methods to attack such recursions, but also as a *catalogue of solutions*, as explained e. g. in [3]: Since solutions are linear, with a suitably large collection of basic solutions one can get ones own solution by constructing an appropriate linear combination.

A good way of imagining what we are doing is by thinking of binary search trees, where each node is weighted by a harmonic number, indexed by the size of the subtree having the particular node as a root. The basic parameter of interest is then the total weight of a binary search tree. (The sum of the weights of all nodes in the binary search tree.)

2. Total weight of a binary search tree

Let $f_n(v)$ be the probability generating function of the parameter "total weight of a binary search tree," i. e. $[v^k]f_n(v)$ is the probability that a random binary search tree of size n has weight k. Observe carefully that this is *not* a polynomial, as the exponents will be in general some rational numbers. That is, however, not a major drawback, although it is harder to set up multivariate generating functions. In this section we are interested in the average $a_n = f_n'(1)$ and variance $b_n + a_n - a_n^2$, with $b_n = f_n''(1)$.

Up to a certain level of complexity, one can resort to elementary methods, as we will demonstrate here. In later sections, however, the appropriate methodology is to work with differential equations.

The recursion

$$f_n(v) = v^{H_n}\frac{1}{n}\sum_{k=0}^{n-1} f_k(v)f_{n-1-k}(v), \qquad n \ge 1,\ f_0(v) = 1$$

is quite straightforward; compare [9] for very similar recursions.

From that we find the prototype of a binary search tree recursion with a harmonic toll function, viz.

$$a_n = H_n + \frac{2}{n}\sum_{k=0}^{n-1} a_k, \qquad n \ge 1,\ a_0 = 0.$$

One can find the solution $a_n = 2n - H_n$ for instance by inspection, which is not really a method but works well in practice when everything is sufficiently simple. Typically, a routine proof by induction would justify the answer. Now set $b_n = f_n''(1)$, then

$$b_n = H_n(H_n - 1) + 2H_n(a_n - H_n) + \frac{2}{n}\sum_{k=0}^{n-1} b_k + \frac{2}{n}\sum_{k=0}^{n-1} a_k a_{n-1-k}, \qquad n \ge 1,\ b_0 = 0.$$

Plugging in the known values and simplifying (we spare the reader with the simple but boring details) we find

$$nb_n - (n+1)b_{n-1} = 4n^2 - H_n^2 - 3H_n + \frac{2}{n}H_n - 2H_n^{(2)} + \frac{1}{n^2} - 1,$$

where also harmonic numbers of the second order occur. The technique is explained in many textbooks on analysis of algorithms, e. g. [9]. The first step is to multiply by n, so that the difference of two such equations, once indexed by n and once by $n-1$, makes the sum $\sum_{k=0}^{n-1} b_k$ disappear.

The recursion may be rewritten as

$$\begin{aligned}\frac{b_n}{n+1}-\frac{b_{n-1}}{n}=4-\frac{4}{n+1}-\frac{H_n^2}{n}+\frac{H_{n+1}^2}{n+1}-\frac{2H_{n+1}}{(n+1)^2}+\frac{2H_n}{n^2}-\frac{5H_n}{n}+\frac{5H_{n+1}}{n+1}\\ -\frac{5}{(n+1)^2}-\frac{2H_n^{(2)}}{n}+\frac{2H_{n+1}^{(2)}}{n+1}-\frac{1}{(n+1)^3}+\frac{1}{n^3}-\frac{1}{n^2},\end{aligned}$$

which can now be summed (a telescoping sum), with the result

$$\frac{b_n}{n+1}=4n-4H_n-\frac{4}{n+1}+4+\frac{H_n^2}{n+1}+5\frac{H_n}{n+1}-6H_n^{(2)}+2\frac{H_n^{(2)}}{n+1},$$

or

$$b_n=4n(n+2)-(4n-1)H_n+H_n^2-2(3n+2)H_n^{(2)}.$$

In the simplification process of these sums, the most important identity is

$$\sum_{k=1}^{n}\binom{k}{m}H_k=\binom{n+1}{m+1}\Big(H_{n+1}-\frac{1}{m+1}\Big),$$

which can be found in [5] and other textbooks.

Thus the variance turns out to be

$$\begin{aligned}b_n+a_n-a_n^2&=4n(n+2)-(4n-1)H_n+H_n^2-2(3n+2)H_n^{(2)}\\&\quad+2n-H_n-4n^2+4nH_n-H_n^2\\&=10n-2(3n+2)H_n^{(2)}.\end{aligned}$$

Asymptotically, the variance is given by

$$(10-\pi^2)n=0.130395596\cdot n.$$

3. Refined results a la quickselect (=Hoare's Find algorithm)

Traditionally, one associates the numbers $1,2,\dots,n$ with the nodes of the binary search tree by traversing it in *inorder*. This reflects also the intuition that smaller elements are to the left, and larger elements to the right of the root. If one uses the quicksort strategy only to *find* element j, then the recursion of interest is

$$P_{n,j}(v)=v^*\frac{1}{n}\Big[\sum_{k=1}^{j-1}P_{n-k,j-k}(v)+1+\sum_{k=j+1}^{n}P_{k-1,j}(v)\Big],\quad P_{1,1}(v)=v,$$

where $*$ stands for 1 in the instance "number of passes," and for $n-1$ in the instance "number of comparisons." See for instance [4] and the references therein.

In the context of our study of harmonic recursions, it is thus naturally to introduce the probability generating functions

$$P_{n,j}(v) = v^{H_n}\frac{1}{n}\Bigg[\sum_{k=1}^{j-1} P_{n-k,j-k}(v) + 1 + \sum_{k=j+1}^{n} P_{k-1,j}(v)\Bigg], \quad P_{1,1}(v) = v.$$

The coefficient of v^k in it is the probability that in a random binary search tree of size n the node j is connected to the root via the path $r_1, r_2, \dots, r_s = j$, and the sum of the weights of the nodes $r_1, r_2, \dots, r_s$ is k.

The expectations are again given by $a_{n,j} = P'_{n,j}(1)$, or

$$a_{n,j} = H_n + \frac{1}{n}\Bigg[\sum_{k=1}^{j-1} a_{n-k,j-k} + \sum_{k=j+1}^{n} a_{k-1,j}\Bigg], \quad a_{1,1} = 1.$$

Although even here is it possible to solve it by elementary means, it is the right time to introduce generating functions,

$$F(z,u) = \sum_{1\le j\le n} a_{n,j} z^n u^j.$$

In order to translate the recursion into a differential equation, one must first compute

$$\sum_{1\le j\le n} nH_n z^{n-1}u^j = \frac{u}{1-u}\big(S(z) - S(zu)\big)$$

with

$$S(z) = \left(\frac{1}{1-z}\log\left(\frac{1}{1-z}\right)\right)' = \frac{1}{(1-z)^2}\log\left(\frac{1}{1-z}\right) + \frac{1}{(1-z)^2}.$$

Hence we obtain the differential equation

$$\frac{\partial}{\partial z}F(z,u) = \frac{u}{1-u}\big(S(z) - S(zu)\big) + \left(\frac{u}{1-zu} + \frac{1}{1-z}\right)F(z,u), \qquad F(0,u) = 0$$

which is well within the range of Maple, and we obtain

$$\begin{aligned} F(z,u) = {} & \frac{1}{2(1-z)(1-zu)}\log^2\frac{1}{1-zu} + \frac{u}{2(1-z)(1-zu)}\log^2\frac{1}{1-z} \\ & + \frac{2-u-zu}{(1-z)(1-zu)(1-u)}\log\frac{1}{1-zu} + \frac{u(1+zu-2u)}{(1-z)(1-zu)(1-u)}\log\frac{1}{1-z} \\ & - \frac{2zu}{(1-zu)(1-z)}. \end{aligned}$$

Reading off coefficients is a (tedious) routine operation, with the result

$$a_{n,j} = \frac{1}{2}H_j^2 + \frac{1}{2}H_{n+1-j}^2 - \frac{1}{2}H_j^{(2)} - \frac{1}{2}H_{n+1-j}^{(2)} - H_n + 2H_j + 2H_{n+1-j} - 2.$$

One should observe the symmetry of the results (obvious from the definition): for each term with j, there is also a corresponding term with $n+1-j$.

To gain some asymptotic information, it is customary in this context to set $j = \rho n$, with a fixed ratio $0 < \rho < 1$. Then $a_{n,j} \sim \log^2 n$.

Observe also that

$$\begin{aligned}\sum_{j=1}^{n} a_{n,j} &= \sum_{j=1}^{n}\Big(H_j^2 - H_j^{(2)}\Big) - nH_n + 4\sum_{j=1}^{n} H_j - 2n \\ &= (n+1)\Big(H_n^2 - H_n^{(2)}\Big) + 2n - 2nH_n - nH_n + 4(n+1)H_n - 4n - 2n \\ &= (n+1)\Big(H_n^2 - H_n^{(2)}\Big) + nH_n + 4H_n - 4n.\end{aligned}$$

Now we turn to the second factorial moments, which are given by $b_{n,j} = P''_{n,j}(1)$. We get here the recurrence

$$b_{n,j} = H_n(H_n-1) + 2\frac{H_n}{n}\left[\sum_{k=1}^{j-1} a_{n-k,j-k} + \sum_{k=j+1}^{n} a_{k-1,j}\right] + \frac{1}{n}\left[\sum_{k=1}^{j-1} b_{n-k,j-k} + \sum_{k=j+1}^{n} b_{k-1,j}\right], \quad b_{1,1} = 0,$$

which simplifies by the above recursion for the $a_{n,j}$'s to

$$\begin{aligned}b_{n,j} &= -3H_n^2 - 5H_n + H_nH_j^2 + H_nH_{n+1-j}^2 - H_nH_j^{(2)} - H_nH_{n+1-j}^{(2)} + 4H_nH_j + 4H_nH_{n+1-j} \\ &\quad + \frac{1}{n}\sum_{k=1}^{j-1} b_{n-k,j-k} + \frac{1}{n}\sum_{k=j+1}^{n} b_{k-1,j}.\end{aligned}$$

The solution can be obtained by the following considerations. The recurrence

$$nb_{n,j} = \sum_{k=1}^{j-1} b_{n-k,j-k} + \sum_{k=j+1}^{n} b_{k-1,j} + X_{n,j}\,,$$

where $X_{n,j}$ is an arbitrary toll function with ordinary generating function $X(z,u)$, leads to the differential equation

$$\frac{\partial}{\partial z}F(z,u) - \left(\frac{1}{1-z} + \frac{u}{1-uz}\right)F(z,u) = \frac{X(z,u)}{z}\,,$$

which has the solution

$$F(z,u) = \frac{1}{(1-z)(1-uz)}\int_{t=0}^{z}\frac{(1-t)(1-ut)}{t}X(t,u)dt\,.$$

(In this case, Maple cannot do that, since the integral is no longer expressible in elementary functions.)

Extracting coefficients gives after some manipulations

$$b_{n,j} = \sum_{k=1}^{j}\sum_{r=k}^{n-j+k}\frac{X_{r,k}}{r} - \sum_{k=1}^{j}\sum_{r=k+1}^{n-j+k}\frac{X_{r-1,k}}{r} - \sum_{k=2}^{j}\sum_{r=k}^{n-j+k}\frac{X_{r-1,k-1}}{r} + \sum_{k=2}^{j}\sum_{r=k+1}^{n-j+k}\frac{X_{r-2,k-1}}{r}\,.$$

In our case, the $X_{n,j}$'s are given by

$$\begin{aligned}X_{n,j} &= -3nH_n^2 - 5nH_n + nH_nH_j^2 + nH_nH_{n+1-j}^2 \\ &\quad - nH_nH_j^{(2)} - nH_nH_{n+1-j}^{(2)} + 4nH_nH_j + 4nH_nH_{n+1-j}\,.\end{aligned}$$

The following observation is also of some interest: If two toll functions $X_{n,j}$ resp. $\bar{X}_{n,j}$ are given by

$$X_{n,j} = C_nD_j \quad \text{and} \quad \bar{X}_{n,j} = C_nD_{n+1-j}\,,$$

then for $b_{n,j}$ resp. $\bar{b}_{n,j}$ we have

$$\bar{b}_{n,j} = b_{n,n+1-j} \, .$$

So the problem is reduced to double summations for the terms nH_n, nH_n^2, nH_nH_j, $nH_nH_j^{(2)}$ and $nH_nH_j^2$. Because of cancellations that occur a simple rewriting of $b_{n,j}$ is advantageous with respect to an easier manipulation of the terms involved:

$$b_{n,j} = \sum_{k=1}^{j} \sum_{r=k+1}^{n-j+k} \frac{1}{r}(X_{r,k} - X_{r-1,k}) - \sum_{k=2}^{j} \sum_{r=k+1}^{n-j+k} \frac{1}{r}(X_{r-1,k-1} - X_{r-2,k-1}) + \sum_{k=2}^{j} \frac{1}{r}(X_{k,k} - X_{k-1,k-1}) + X_{1,1}.$$

For all the terms noted above, we can substitute the inner sums by expressions involving harmonic numbers, and simplify the remaining sum using formulæ like

$$\sum_{k=1}^{n} \frac{H_k}{k} = \frac{1}{2}H_n^2 + \frac{1}{2}H_n^{(2)} \, ,$$

$$\sum_{k=1}^{n} \frac{H_k^{(2)}}{k^2} = \frac{1}{2}(H_n^{(2)})^2 + \frac{1}{2}H_n^{(4)} \, ,$$

$$\sum_{k=1}^{n} \frac{H_k^2}{k} = \frac{1}{3}H_n^3 + H_nH_n^{(2)} + \frac{2}{3}H_n^{(3)} - \sum_{k=1}^{n} \frac{H_k^{(2)}}{k} \, ,$$

$$\sum_{k=1}^{j} \frac{H_{n-j+k}}{k} + \sum_{k=1}^{j} \frac{H_k}{n-j+k} = H_nH_j + \frac{1}{n-j}\left(H_j + H_{n-j} - H_n\right) \, ,$$

$$\begin{aligned} \sum_{k=1}^{j} \frac{H_k^2}{n-j+k} &= \sum_{k=1}^{j} \frac{H_{n-j+k}}{k^2} - 2\sum_{k=1}^{j} \frac{H_kH_{n-j+k}}{k} + \frac{2}{n-j}\sum_{k=1}^{j} \frac{H_{n-j+k}}{k} \\ &\quad + \frac{H_j^2}{n-j} + H_nH_j^2 - \frac{2H_nH_j}{n-j} - \frac{1}{(n-j)^2}\left(H_j + H_{n-j} - H_n\right) \, . \end{aligned}$$

Eventually we find

$$\begin{aligned} b_{n,j} &= \sum_{k=1}^{j} \frac{H_kH_{n-j+k}^2}{k} - \sum_{k=1}^{j} \frac{H_kH_{n-j+k}^{(2)}}{k} + 4\sum_{k=1}^{j} \frac{H_kH_{n-j+k}}{k} - \sum_{k=1}^{j} \frac{H_{n-j+k}^2}{k^2} + \sum_{k=1}^{j} \frac{H_{n-j+k}^{(2)}}{k^2} \\ &\quad + 2\sum_{k=1}^{j} \frac{H_{n-j+k}^2}{k} - 4\sum_{k=1}^{j} \frac{H_{n-j+k}}{k^2} - 2\sum_{k=1}^{j} \frac{H_{n-j+k}^{(2)}}{k} + 8\sum_{k=1}^{j} \frac{H_{n-j+k}}{k} \\ &\quad + \sum_{k=1}^{n-j+1} \frac{H_kH_{j-1+k}^2}{k} - \sum_{k=1}^{n-j+1} \frac{H_kH_{j-1+k}^{(2)}}{k} + 4\sum_{k=1}^{n-j+1} \frac{H_kH_{j-1+k}}{k} - \sum_{k=1}^{n-j+1} \frac{H_{j-1+k}^2}{k^2} + \sum_{k=1}^{n-j+1} \frac{H_{j-1+k}^{(2)}}{k^2} \\ &\quad + 2\sum_{k=1}^{n-j+1} \frac{H_{j-1+k}^2}{k} - 4\sum_{k=1}^{n-j+1} \frac{H_{j-1+k}}{k^2} - 2\sum_{k=1}^{n-j+1} \frac{H_{j-1+k}^{(2)}}{k} + 8\sum_{k=1}^{n-j+1} \frac{H_{j-1+k}}{k} \\ &\quad - 2\sum_{k=1}^{j} \frac{H_k^2}{k^2} - 4\sum_{k=1}^{j} \frac{H_k^{(3)}}{k} + 5\sum_{k=1}^{j} \frac{H_k^{(2)}}{k} - 2\sum_{k=1}^{n-j+1} \frac{H_k^2}{k^2} - 4\sum_{k=1}^{n-j+1} \frac{H_k^{(3)}}{k} + 5\sum_{k=1}^{n-j+1} \frac{H_k^{(2)}}{k} \end{aligned}$$

$$+\frac{1}{3}H_j^3 - 5H_jH_j^{(2)} - \frac{1}{3}H_j^{(3)} - \frac{9}{2}H_j^2 + \frac{12j+9}{2}H_j^{(2)} + 2H_j^{(4)} - 22H_j + 4H_jH_j^{(3)}$$
$$+\frac{1}{3}H_{n-j+1}^3 - 5H_{n-j+1}H_{n-j+1}^{(2)} - \frac{1}{3}H_{n-j+1}^{(3)} - \frac{9}{2}H_{n-j+1}^2 + \frac{12(n-j+1)+9}{2}H_{n-j+1}^{(2)}$$
$$+ 2H_{n-j+1}^{(4)} - 22H_{n-j+1} + 4H_{n-j+1}H_{n-j+1}^{(3)}$$
$$- H_j^2H_n + H_j^{(2)}H_n - 4H_jH_n - H_{n-j+1}^2H_n + H_{n-j+1}^{(2)}H_n - 4H_{n-j+1}H_n$$
$$+ 3H_n^2 - 6(n+1)H_n^{(2)} + 17H_n + 16.$$

To find an asymptotic equivalent for $b_{n,j}$ for a fixed ratio $j = \rho n$, with $0 < \rho < 1$ and $n \to \infty$ we can use the following relations, which can be obtained by using the asymptotic expansion $H_{n-j+k} = \log n + \log(1-\rho) + \gamma + \log\left(1 + \frac{k}{(1-\rho)n}\right) + \mathcal{O}\left(\frac{1}{n}\right)$ for $1 \le k \le j = \rho n$:

$$\sum_{k=1}^{j} \frac{H_kH_{n-j+k}}{k} = \frac{1}{2}\log^3 n + \mathcal{O}\left(\log^2 n\right) = \sum_{k=1}^{n-j+1} \frac{H_kH_{j-1+k}}{k},$$
$$\sum_{k=1}^{j} \frac{H_kH_{n-j+k}^2}{k} = \frac{1}{2}\log^4 n + (\log\rho + \log(1-\rho) + 2\gamma)\log^3 n + \mathcal{O}\left(\log^2 n\right) = \sum_{k=1}^{n-j+1} \frac{H_kH_{j-1+k}^2}{k},$$
$$\sum_{k=1}^{j} \frac{H_{n-j+k}^2}{k} = \log^3 n + \mathcal{O}\left(\log^2 n\right) = \sum_{k=1}^{n-j+1} \frac{H_{j-1+k}^2}{k}.$$

The assumption that $j = \rho n$ with $0 < \rho < 1$, is a usual one in the analysis of algorithms. (For instance in the study of hashing algorithms such a proportionality factor as ρ is called the *load factor*.) One particular aspect of such an assumption is that one has to consider only one parameter (n) which goes to infinity. Of course, such an assumption does not necessarily cover instances like $n, j \to \infty$, as $j \sim \log n$.

The other terms involving sums are bounded by $\mathcal{O}\left(\log^2 n\right)$.

This leads to the asymptotic expansion for the second factorial moments $b_{n,j}$,

$$b_{n,j} = \log^4 n + \left(2\log\rho + 2\log(1-\rho) + 4\gamma + \frac{20}{3}\right)\log^3 n + \mathcal{O}\left(\log^2 n\right),$$

for $j = \rho n$ with a fixed $0 < \rho < 1$. As a consequence we find as the asymptotic equivalent of the variance $b_{n,j} + a_{n,j} - a_{n,j}^2 \sim \frac{2}{3}\log^3 n$.

4. Weight of subtrees

We consider now the total weight of subtrees in binary search trees. To do this, we introduce the probability generating function $P_{n,j}(v)$ of the total weight of the subtree j in a binary search tree with n nodes. In other words, the coefficient $[v^k]P_{n,j}(v)$ is the probability that the total weight of the subtree with root j is k, assuming that all permutations of n elements are equally likely. The average $a_{n,j}$ is then obtained (as usual) as $a_{n,j} = P'_{n,j}(1)$, and the variance as $b_{n,j} + a_{n,j} - \left(a_{n,j}\right)^2$, with $b_{n,j} = P''_{n,j}(1)$.

We get the recurrence

$$P_{n,j}(v) = \frac{1}{n}\left(\sum_{k=1}^{j-1} P_{n-k,j-k}(v) + \sum_{k=j+1}^{n} P_{k-1,j}(v) + v^{H_n} f_{j-1}(v) f_{n-j}(v)\right), \quad P_{1,1}(v) = v,$$

where $f_n(v)$ is the probability generating function of the total weight of a binary search tree, as introduced in section 2. This recursion is obtained by considering the three cases that the subtree with root j is in the right subtree or left subtree, or that it is the whole tree, in which case we must add a contribution of H_n to the total count.

For the expectations $a_{n,j} = P'_{n,j}(1)$, we get the following recursion:

$$a_{n,j} = \frac{1}{n}\sum_{k=1}^{j-1} a_{n-k,j-k} + \frac{1}{n}\sum_{k=j+1}^{n} a_{k-1,j} + \frac{1}{n}(2n + H_n - H_{j-1} - H_{n-j} - 2)\,, \quad a_{1,1} = 1\,.$$

An approach with generating functions leads to the result

$$a_{n,j} = (n+1)H_n^{(2)} - jH_j^{(2)} - (n+1-j)H_{n+1-j}^{(2)} + 2H_j + 2H_{n+1-j} + \frac{1}{j} + \frac{1}{n+1-j} - 5\,,$$

what means that the weight of the subtree j with $j = \rho n$ and $0 < \rho < 1$ is asymptotically $4\log n$.

5. A catalogue of solutions

In this section we give a list of solutions of

$$a_{n,j} = \textsf{toll-function} + \frac{1}{n}\Bigg[\sum_{k=1}^{j-1} a_{n-k,j-k} + \sum_{k=j+1}^{n} a_{k-1,j}\Bigg], \quad a_{1,1} = 1.$$

with various toll functions of the form $n^d H_n$ and (simpler) n^d. We write $\mathbf{j} = n+1-j$ for convenience. (The solutions are obtained by an *ansatz* and matching coefficients.) Such a catalogue is especially useful for a *repertoire approach,* see [3].

For H_n we have

$$a_{n,j} = \frac{1}{2}\Big(H_j^2 - H_j^{(2)}\Big) + \frac{1}{2}\Big(H_{\mathbf{j}}^2 - H_{\mathbf{j}}^{(2)}\Big) + 2H_j + 2H_{\mathbf{j}} - H_n - 2.$$

For nH_n we have

$$\begin{aligned} a_{n,j} &= (n+1)\Big(H_n^2 - H_n^{(2)}\Big) - (\tfrac{3}{2}+j)\Big(H_j^2 - H_j^{(2)}\Big) - (\tfrac{3}{2}+\mathbf{j})\Big(H_{\mathbf{j}}^2 - H_{\mathbf{j}}^{(2)}\Big) \\ &\quad + (j-4)H_j + (\mathbf{j}-4)H_{\mathbf{j}} + (n+4)H_n - n + 3. \end{aligned}$$

For n^2H_n we have

$$\begin{aligned} a_{n,j} &= -3(n+1)\Big(H_n^2 - H_n^{(2)}\Big) + (\tfrac{7}{2}+3j)\Big(H_j^2 - H_j^{(2)}\Big) + (\tfrac{7}{2}+3\mathbf{j})\Big(H_{\mathbf{j}}^2 - H_{\mathbf{j}}^{(2)}\Big) \\ &\quad - (\tfrac{3}{2}j^2 + \tfrac{7}{2}j - 8)H_j - (\tfrac{3}{2}\mathbf{j}^2 + \tfrac{7}{2}\mathbf{j} - 8)H_{\mathbf{j}} + (n+2)(3n-4)H_n \\ &\quad - \tfrac{7}{2}j\mathbf{j} - \tfrac{1}{4}n^2 + \tfrac{19}{4}n - 3. \end{aligned}$$

For n^3H_n we have

$$\begin{aligned} a_{n,j} &= 7(n+1)\Big(H_n^2 - H_n^{(2)}\Big) - (\tfrac{15}{2}+7j)\Big(H_j^2 - H_j^{(2)}\Big) - (\tfrac{15}{2}+7\mathbf{j})\Big(H_{\mathbf{j}}^2 - H_{\mathbf{j}}^{(2)}\Big) \\ &\quad - (\tfrac{2}{3}j^3 - 5j^2 - \tfrac{26}{3}j + 16)H_j - (\tfrac{2}{3}\mathbf{j}^3 - 5\mathbf{j}^2 - \tfrac{26}{3}\mathbf{j} + 16)H_{\mathbf{j}} + (2n^3 - 6n^2 - 7n + 16)H_n \\ &\quad - (\tfrac{7}{6}n - \tfrac{61}{6})j\mathbf{j} - \tfrac{1}{9}n^3 + \tfrac{3}{2}n^2 - \tfrac{205}{18}n + 3. \end{aligned}$$

For n^4H_n we have

$$\begin{aligned}
a_{n,j} &= -15(n+1)\Big(H_n^2 - H_n^{(2)}\Big) + (\tfrac{31}{2} + 15j)\Big(H_j^2 - H_j^{(2)}\Big) + (\tfrac{31}{2} + 15\mathbf{j})\Big(H_{\mathbf{j}}^2 - H_{\mathbf{j}}^{(2)}\Big) \\
&\quad - (\tfrac{5}{12}j^4 - \tfrac{5}{2}j^3 + \tfrac{145}{12}j^2 + 19j - 32)H_j - (\tfrac{5}{12}\mathbf{j}^4 - \tfrac{5}{2}\mathbf{j}^3 + \tfrac{145}{12}\mathbf{j}^2 + 19\mathbf{j} - 32)H_{\mathbf{j}} \\
&\quad + (\tfrac{5}{3}n^4 - \tfrac{10}{3}n^3 + \tfrac{40}{3}n^2 + \tfrac{52}{3}n - 32)H_n \\
&\quad + \tfrac{23}{72}j^2\mathbf{j}^2 - (\tfrac{23}{36}n^2 - \tfrac{187}{72}n + \tfrac{214}{9})j\mathbf{j} - \tfrac{1}{16}n^4 + \tfrac{19}{24}n^3 - \tfrac{51}{16}n^2 + \tfrac{599}{24}n - 3.
\end{aligned}$$

For n^5H_n we have

$$\begin{aligned}
a_{n,j} &= 31(n+1)\Big(H_n^2 - H_n^{(2)}\Big) - (\tfrac{63}{2} + 31j)\Big(H_j^2 - H_j^{(2)}\Big) - (\tfrac{63}{2} + 31\mathbf{j})\Big(H_{\mathbf{j}}^2 - H_{\mathbf{j}}^{(2)}\Big) \\
&\quad - (\tfrac{3}{10}j^5 - \tfrac{7}{4}j^4 + \tfrac{19}{3}j^3 - \tfrac{105}{4}j^2 - \tfrac{1189}{30}j - 64)H_j \\
&\quad - (\tfrac{3}{10}\mathbf{j}^5 - \tfrac{7}{4}\mathbf{j}^4 + \tfrac{19}{3}\mathbf{j}^3 - \tfrac{105}{4}\mathbf{j}^2 - \tfrac{1189}{30}\mathbf{j} - 64)H_{\mathbf{j}} \\
&\quad + (\tfrac{3}{2}n^5 - \tfrac{5}{2}n^4 + \tfrac{15}{2}n^3 - \tfrac{55}{2}n^2 - 38n + 64)H_n \\
&\quad + (\tfrac{17}{40}n - \tfrac{13}{20})j^2\mathbf{j}^2 - (\tfrac{17}{40}n^3 - \tfrac{7}{8}n^2 + \tfrac{733}{120}n - \tfrac{3053}{60})j\mathbf{j} \\
&\quad - \tfrac{1}{25}n^5 + \tfrac{41}{80}n^4 - \tfrac{497}{360}n^3 + \tfrac{559}{80}n^2 - \tfrac{93743}{1800}n + 3.
\end{aligned}$$

For n^6H_n we have

$$\begin{aligned}
a_{n,j} &= -63(n+1)\Big(H_n^2 - H_n^{(2)}\Big) + (\tfrac{127}{2} + 63j)\Big(H_j^2 - H_j^{(2)}\Big) + (\tfrac{127}{2} + 63\mathbf{j})\Big(H_{\mathbf{j}}^2 - H_{\mathbf{j}}^{(2)}\Big) \\
&\quad - (\tfrac{7}{30}j^6 - \tfrac{7}{5}j^5 + \tfrac{14}{3}j^4 - 14j^3 + \tfrac{273}{5}j^2 + \tfrac{809}{10}j - 128)H_j \\
&\quad - (\tfrac{7}{30}\mathbf{j}^6 - \tfrac{7}{5}\mathbf{j}^5 + \tfrac{14}{3}\mathbf{j}^4 - 14\mathbf{j}^3 + \tfrac{273}{5}\mathbf{j}^2 + \tfrac{809}{10}\mathbf{j} - 128)H_{\mathbf{j}} \\
&\quad + (\tfrac{7}{5}n^6 - \tfrac{21}{10}n^5 + \tfrac{35}{6}n^4 - \tfrac{91}{6}n^3 + \tfrac{1673}{30}n^2 + \tfrac{1189}{15}n - 128)H_n \\
&\quad - \tfrac{47}{450}j^3\mathbf{j}^3 + (\tfrac{47}{100}n^2 - \tfrac{301}{600}n + \tfrac{887}{600})j^2\mathbf{j}^2 \\
&\quad - (\tfrac{47}{150}n^4 - \tfrac{113}{600}n^3 + \tfrac{491}{200}n^2 - \tfrac{3839}{300}n + \tfrac{94553}{900})j\mathbf{j} \\
&\quad - \tfrac{1}{36}n^6 + \tfrac{37}{100}n^5 - \tfrac{241}{360}n^4 + \tfrac{31}{10}n^3 - \tfrac{5239}{360}n^2 + \tfrac{2657}{25}n - 3.
\end{aligned}$$

For n^7H_n we have

$$\begin{aligned}
a_{n,j} &= 127(n+1)\Big(H_n^2 - H_n^{(2)}\Big) - (\tfrac{255}{2} + 127j)\Big(H_j^2 - H_j^{(2)}\Big) - (\tfrac{255}{2} + 127\mathbf{j})\Big(H_{\mathbf{j}}^2 - H_{\mathbf{j}}^{(2)}\Big) \\
&\quad - (\tfrac{4}{21}j^7 - \tfrac{6}{5}j^6 + \tfrac{59}{15}j^5 - \tfrac{21}{2}j^4 + \tfrac{88}{3}j^3 - \tfrac{1113}{10}j^2 - \tfrac{5721}{35}j + 256)H_j \\
&\quad - (\tfrac{4}{21}\mathbf{j}^7 - \tfrac{6}{5}\mathbf{j}^6 + \tfrac{59}{15}\mathbf{j}^5 - \tfrac{21}{2}j^4 + \tfrac{88}{3}\mathbf{j}^3 - \tfrac{1113}{10}\mathbf{j}^2 - \tfrac{5721}{35}\mathbf{j} + 256)H_{\mathbf{j}} \\
&\quad + (\tfrac{4}{3}n^7 - \tfrac{28}{15}n^6 + \tfrac{77}{15}n^5 - \tfrac{35}{3}n^4 + \tfrac{91}{3}n^3 - \tfrac{1687}{15}n^2 - \tfrac{809}{5}n + 256)H_n \\
&\quad - (\tfrac{31}{126}n - \tfrac{347}{3150})j^3\mathbf{j}^3 + (\tfrac{31}{63}n^3 - \tfrac{19}{150}n^2 + \tfrac{1237}{900}n - \tfrac{9839}{3150})j^2\mathbf{j}^2 \\
&\quad - (\tfrac{31}{126}n^5 + \tfrac{509}{3150}n^4 + \tfrac{129}{100}n^3 - \tfrac{30697}{6300}n^2 + \tfrac{164677}{6300}n - \tfrac{112079}{525})j\mathbf{j} \\
&\quad - \tfrac{1}{49}n^7 + \tfrac{2}{7}n^6 - \tfrac{54}{175}n^5 + \tfrac{457}{280}n^4 - \tfrac{389}{60}n^3 + \tfrac{8283}{280}n^2 - \tfrac{3155909}{14700}n + 3.
\end{aligned}$$

And here is the simpler set of solutions:
For 1,

$$a_{n,j} = H_j + H_{\mathbf{j}} - 1,$$

for n,

$$a_{n,j} = -(3j+2)H_j - (3\mathbf{j}+2)H_{\mathbf{j}} + 2(n+1)H_n + 2n + 5$$

for n^2,

$$a_{n,j} = (6j+7)H_j + (6\mathbf{j}+7)H_{\mathbf{j}} - 6(n+1)H_n + \tfrac{3}{2}n^2 - \tfrac{9}{2}n - 13 - \tfrac{9}{2}n + 3j\mathbf{j} - 13$$

for n^3,

$$a_{n,j} = -(14j+15)H_j - (14\mathbf{j}+15)H_{\mathbf{j}} + 14(n+1)H_n + (2n-8)j\mathbf{j} + \tfrac{4}{3}n^3 - 3n^2 + \tfrac{29}{3}n + 29$$

for n^4,

$$\begin{aligned} a_{n,j} = {} & (30j+31)H_j + (30\mathbf{j}+31)H_{\mathbf{j}} - 30(n+1)H_n \\ & + (\tfrac{5}{3}n^2 - \tfrac{25}{6}n + \tfrac{55}{3})j\mathbf{j} + \tfrac{5}{4}n^4 - \tfrac{5}{2}n^3 + \tfrac{25}{4}n^2 - 20n - 61 - \tfrac{5}{6}j^2\mathbf{j}^2 \end{aligned}$$

for n^5,

$$\begin{aligned} a_{n,j} = {} & -(62j+63)H_j - (62\mathbf{j}+63)H_{\mathbf{j}} + 62(n+1)H_n \\ & - (\tfrac{3}{2}n - 2)j^2\mathbf{j}^2 + (\tfrac{3}{2}n^3 - \tfrac{5}{2}n^2 + \tfrac{19}{2}n - 39)j\mathbf{j} + \tfrac{6}{5}n^5 - \tfrac{9}{4}n^4 + \tfrac{31}{6}n^3 - \tfrac{51}{4}n^2 + \tfrac{1219}{30}n + 125 \end{aligned}$$

for n^6,

$$\begin{aligned} a_{n,j} = {} & (126j+127)H_j + (126\mathbf{j}+127)H_{\mathbf{j}} - 126(n+1)H_n \\ & + \tfrac{7}{15}j^3\mathbf{j}^3 - (\tfrac{21}{10}n^2 - \tfrac{14}{5}n + \tfrac{133}{30})j^2\mathbf{j}^2 + (\tfrac{7}{5}n^4 - \tfrac{7}{5}n^3 + \tfrac{91}{15}n^2 - \tfrac{301}{15}n + \tfrac{1204}{15})j\mathbf{j} \\ & + \tfrac{7}{6}n^6 - \tfrac{21}{10}n^5 + \tfrac{14}{3}n^4 - \tfrac{21}{2}n^3 + \tfrac{77}{3}n^2 - \tfrac{819}{10}n - 253, \end{aligned}$$

for n^7,

$$\begin{aligned} a_{n,j} = {} & -(254j+255)H_j - (254\mathbf{j}+255)H_{\mathbf{j}} + 254(n+1)H_n \\ & + (\tfrac{4}{3}n - \tfrac{16}{15})j^3\mathbf{j}^3 - (\tfrac{8}{3}n^3 - \tfrac{14}{5}n^2 + \tfrac{91}{15}n - \tfrac{142}{15})j^2\mathbf{j}^2 \\ & + (\tfrac{4}{3}n^5 - \tfrac{8}{15}n^4 + \tfrac{21}{5}n^3 - \tfrac{193}{15}n^2 + \tfrac{613}{15}n - \tfrac{814}{5})j\mathbf{j} \\ & + \tfrac{8}{7}n^7 - 2n^6 + \tfrac{22}{5}n^5 - \tfrac{19}{2}n^4 + 21n^3 - \tfrac{103}{2}n^2 + \tfrac{5756}{35}n + 509. \end{aligned}$$

(Several general formulæ for the coefficients can be spotted from that.)

6. Results a la Multiple Quickselect

Several authors have considered *Multiple Quickselect*; see [6, 8, 7]. The idea is to use Quicksort (vs. Quickselect) to find a prescribed set of values, e. g. the third and the tenth; in general we search simultaneously for p elements with ranks $j_1, j_2, \dots, j_p$ where $1 \le j_1 < j_2 < \dots < j_p \le n$. Here we consider analogous recursions with harmonic toll functions:

$$X[n; j_1, \dots, j_p] = H_n + \frac{1}{n}\Bigg(\sum_{t=1}^{p-1} \sum_{k=j_t+1}^{j_{t+1}} X[k-1; j_1, \dots, j_t] + \sum_{k=j_p+1}^{n} X[k-1; j_1, \dots, j_p] \\ + \sum_{k=1}^{j_1-1} X[n-k; j_1-k, \dots, j_p-k] + \sum_{t=1}^{p-1} \sum_{k=j_t}^{j_{t+1}-1} X[n-k; j_{t+1}-k, \dots, j_p-k] \Bigg).$$

One finds that the solution of this recursion is given by

$$X[n; j_1, \dots, j_p] = \frac{1}{2}H_{j_1}^2 - \frac{1}{2}H_{j_1}^{(2)} + 2H_{j_1} + \frac{1}{2}H_{n+1-j_p}^2 - \frac{1}{2}H_{n+1-j_p}^{(2)} + 2H_{n+1-j_p} \\ + \sum_{t=2}^{p} \Big(H_{j_t+1-j_{t-1}}^2 - H_{j_t+1-j_{t-1}}^{(2)} + 2H_{j_t+1-j_{t-1}} \Big) - H_n - 2p\,.$$

The proof can be done by induction. The only formulæ which are required, are

$$\sum_{k=1}^{n} H_k^2 = (n+1)H_n^2 - (2n+1)H_n + 2n\,, \qquad \sum_{k=1}^{n} H_k^{(2)} = (n+1)H_n^{(2)} - H_n\,,$$

$$\sum_{k=1}^{n} H_k = (n+1)H_n - n\,.$$

The proof is omitted here, because it is long, but completely routine. It should be remarked, that for the case $p = 2$ the recursion was solved by setting up generating functions and extracting coefficients of the solution of the resulting linear differential equation. The solution of the general case for arbitrary p was then guessed and verified by computer experiments.

7. Results a la Grand Averages

For the analysis of Multiple Quickselect, it was suggested that, instead of a given set of p values (see [8]) it is easier to consider all $\binom{n}{p}$ subsets of size p. This leads to the *Grand Averages* of the number of passes and the number of comparisons. Here again we study such questions together with harmonic toll functions.

With the notation from the previous section, this means that we are interested in

$$\mathcal{X}_{n,p} = \frac{1}{\binom{n}{p}} \sum_{1 \le j_1 < j_2 < \dots < j_p \le n} X[n; j_1, \dots, j_p]\,.$$

To evaluate this, we use two summation formulæ, which can be found in [8]:

$$\sum_{1 \le j_1 < j_2 < \dots < j_p \le n} H_{j_1} = \binom{n+1}{p}(H_{n+1} - H_p) = \sum_{1 \le j_1 < j_2 < \dots < j_p \le n} H_{n+1-j_p}\,,$$

$$\sum_{t=2}^{p} \sum_{1 \le j_1 < j_2 < \dots < j_p \le n} H_{j_t+1-j_{t-1}} = (p-1)\binom{n+2}{p}(H_{n+2} - H_p) - (p-1)\binom{n}{p-1}\,.$$

In a similar manner as shown there, one gets the remaining required formulæ:

$$\sum_{1\le j_1<j_2<\cdots<j_p\le n}(H_{j_1}^2-H_{j_1}^{(2)})=\binom{n+1}{p}\Big((H_{n+1}-H_p)^2-(H_{n+1}^{(2)}-H_p^{(2)})\Big)$$
$$=\sum_{1\le j_1<j_2<\cdots<j_p\le n}(H_{n+1-j_p}^2-H_{n+1-j_p}^{(2)}),$$

$$\sum_{t=2}^{p}\sum_{1\le j_1<j_2<\cdots<j_p\le n}(H_{j_t+1-j_{t-1}}^2-H_{j_t+1-j_{t-1}}^2)=(p-1)\binom{n+2}{p}\Big((H_{n+2}-H_p)^2-(H_{n+2}^{(2)}-H_p^{(2)})\Big).$$

Collecting all contributions and dividing by $\binom{n}{p}$, we get for the grand average:

$$\begin{aligned}\mathcal{X}_{n,p}&=\frac{p(n+1)^2}{(n+2-p)(n+1-p)}\Big((H_n-H_p)^2-H_n^{(2)}+H_p^{(2)}\Big)\\&+\frac{(1+2p)n^2+(1+8p)n+p(7-p)}{(n+2-p)(n+1-p)}H_n\\&-\frac{2(n+1)\big((p+1)n+2p+1\big)}{(n+2-p)(n+1-p)}H_p-\frac{2p(n^2-pn-2)}{(n+2-p)(n+1-p)}.\end{aligned}$$

The formula for $\mathcal{X}_{n,p}$ can also be obtained more directly with generating functions. The starting point for this is the following recursion:

$$\psi_n(u,v)=v^{H_n}\frac{1+u}{n}\sum_{j=0}^{n-1}\psi_j(u,v)\psi_{n-1-j}(u,v)+1-v^{H_n},\qquad \psi_0(u,v)=1.$$

Each $[u^p]\psi_n(u,v)/\binom{n}{p}$ is a probability generating function, namely the coefficient of v^k in it is the probability that the smallest subtree of a random binary search tree of size n containing the root and a random sample of p nodes has total weight k. The factor $1+u$ reflects the fact that the root can be in the sample or not, and the extra terms correspond to the case of $p=0$; in this case the total weight must be 0.

If we denote

$$\sigma_n(u)=\frac{\partial}{\partial v}\psi_n(u,1),$$

then

$$\sigma_n(u)=H_n(1+u)^n+\frac{2(1+u)}{n}\sum_{j=0}^{n-1}\sigma_j(u)\sigma_{n-1-j}(u).$$

This recursion can again be translated into a differential equation, viz.

$$\frac{\partial}{\partial z}G(z,u)=(1+u)S\big(z(1+u)\big)-S(z)+\frac{2(1+u)}{1-z(1+u)}G(z,u),\qquad G(0,u)=0,$$

again with $S(z)=\frac{1}{(1-z)^2}\log\frac{1}{1-z}+\frac{1}{(1-z)^2}$. The solution is

$$\begin{aligned}G(z,u)&=\frac{u(1+u)}{(1-z(1+u))^2}\log^2\frac{1}{1-z}\\&-\frac{(1+u)^2z^2-2(1+u)(2u+1)z+2u^2+4u+1}{(1-z)(1-z(1+u))^2}\log\frac{1}{1-z}\\&-\frac{1}{1-z(1+u)}\log\frac{1}{1-z(1+u)}-\frac{2u(1+u)z}{(1-z(1+u))^2}.\end{aligned}$$

It is again a routine task to read off the coefficients, and we get for $p\ge 1$

$$\begin{aligned}[u^p]\sigma_n(u)&=p\binom{n+1}{p}\Big[(H_{n+1}-H_p)^2-(H^{(2)}_{n+1}-H^{(2)}_p)\Big]\\&+(p-1)\binom{n+1}{p-1}\Big[(H_{n+1}-H_{p-1})^2-(H^{(2)}_{n+1}-H^{(2)}_{p-1})\Big]\\&+2(p-1)\binom{n+2}{p}\big[H_{n+2}-H_p\big]+4\binom{n+1}{p}\big[H_{n+1}-H_p\big]\binom{n}{p}H_n-2\binom{n}{p-1}n.\end{aligned}$$

In order to get the desired averages, one must divide that by $\binom{n}{p}$, which yields after simplifications

$$\begin{aligned}\frac{[u^p]\sigma_n(u)}{\binom{n}{p}}=\mathcal{X}_{n,p}&=\frac{p(n+1)^2}{(n+2-p)(n+1-p)}\Big((H_n-H_p)^2-H^{(2)}_n+H^{(2)}_p\Big)\\&+\frac{(1+2p)n^2+(1+8p)n+p(7-p)}{(n+2-p)(n+1-p)}H_n\\&-\frac{2(n+1)\big((p+1)n+2p+1\big)}{(n+2-p)(n+1-p)}H_p-\frac{2p(n^2-pn-2)}{(n+2-p)(n+1-p)}.\end{aligned}$$

For fixed p (close to the Quickselect instance), this is roughly $p\log^2 n$; for $p\sim n$ (close to the Quicksort instance), this is roughly $2n$, and for $p\sim\rho n$, with $0<\rho<1$ it is asymptotic to $\big(\frac{\rho}{(1-\rho)^2}\log^2\rho-\frac{2\rho}{(1-\rho)^2}\log\rho-\frac{2\rho}{1-\rho}\big)n$. The limit of this for $\rho\to 1$ is $2n$, which matches the intuition.

It is interesting to note the special instance $p=1$:

$$\frac{n+1}{n}(H_n^2-H_n^{(2)})+\frac{n+4}{n}H_n-4.$$

Note that this is the same result as we obtained earlier for

$$\frac{1}{n}\sum_{j=1}^{n}a_{n,j},$$

as it should. The other interesting special case is $p=n$:

$$2n-H_n.$$

This is also clear, since there is no random sample involved, and only the average total weight of a random binary search tree has been computed.

References

[1] J. Fill. On the distribution of binary search trees under the random permutation model. *Random Structures and Algorithms*, 8:1–25, 1996.

[2] P. Grabner and H. Prodinger. Sorting algorithms for broadcast communications: Mathematical analysis. *Theoretical Computer Science, to appear*, 2000.

[3] D. H. Greene and D. E. Knuth. *Mathematics for the analysis of algorithms.* Birkhauser, Boston, second edition, 1982.

[4] P. Kirschenhofer and H. Prodinger. Comparisons in Hoare's Find algorithm. *Combinatorics, Probability, and Computing*, 7:111–120, 1998.

[5] D. E. Knuth. *The Art of Computer Programming*, volume 1: Fundamental Algorithms. Addison-Wesley, 1968. Third edition, 1997.

[6] J. Lent and H. M. Mahmoud. Average–case analysis of multiple quickselect: An algorithm for finding order statistics. *Statistics and Probability Letters*, 28:299–310, 1996.

[7] A. Panholzer and H. Prodinger. A generating functions approach for the analysis of grand averages for multiple quickselect. *Random Structures and Algorithms*, 1999.

[8] H. Prodinger. Multiple quickselect — Hoare's Find algorithm for several elements. *Information Processing Letters*, 56:123–129, 1995.

[9] R. Sedgewick and P. Flajolet. *An Introduction to the Analysis of Algorithms.* Addison-Wesley, 1996.

†Institut für Algebra und Computermathematik, Technische Universität Wien, Wiedner Hauptstrasse 8–10, A- 1040 Wien, Austria.

E-mail address: Alois.Panholzer@tuwien.ac.at

‡The John Knopfmacher Centre for Applicable Analysis and Number Theory, Mathematics Department, University of the Witwatersrand, P.O. Wits, 2050 Johannesburg, South Africa.

E-mail address: helmut@gauss.cam.wits.ac.za

WWW-address: http://www.wits.ac.za/helmut/index.htm