THE LEVEL OF NODES IN HEAP ORDERED TREES

HELMUT PRODINGER

Abstract. A heap ordered tree with n nodes ("size n") is a planted plane tree together with a bijection from the nodes to the set $\{1, \dots, n\}$ which is monotonically increasing when going from the root to the leaves. We consider the level of the node j in a (random) heap ordered tree of size $n \geq j$. This distribution does not depend on n. Precise expressions are derived for the expectation, the variance, and the probability distribution.

1. Heap ordered trees

A *heap ordered tree* with n nodes ("size n") might be described as a *planted plane tree* together with a bijection from the nodes to the set $\{1, \dots, n\}$ which is *monotonically increasing* when going from the root to the leaves.

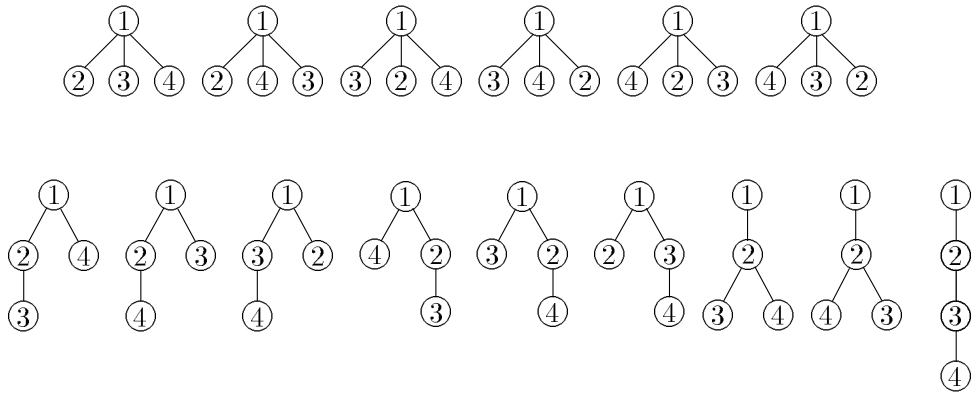

Figure 1. All 15 heap ordered trees with 4 nodes

In this note, we want to concentrate on the level of the node j in a (random) *heap ordered tree* of size $n \geq j$. This extends some of the results in [9] and [2] where the average over j (the *depth of a random node*) was considered. It will turn out, that surprisingly, this distribution does not depend on n. Precise expressions are derived for the expectation, the variance, and the probability distribution. A general reference for heap ordered trees and, more generally, *increasing* trees is [1].

The enumeration of the numbers a_n of heap ordered trees of size n is easy and appears already in [10]. The recursion is

$$a_{n+1} = \sum_{m \geq 1} \sum_{h_1 + \cdots + h_m = n} \binom{n}{h_1, \dots, h_m} a_{h_1} \dots a_{h_m} \quad \text{for } n \geq 1, \quad a_1 = 1 \; ; \tag{1}$$

Date: May 7, 1996.

hence the exponential generating function $A(z) := \sum_{n\ge0} a_n \frac{z^n}{n!}$ fulfills the differential equation

$$A'(z) = \frac{1}{1-A(z)} \qquad \text{with} \quad A(0)=0\ ,$$

with the solution

$$A(z) = 1-\sqrt{1-2z}\ ,$$

so that

$$a_n = n!\,2^{1-n}\,\mathcal{C}_n = 1\cdot3\cdot5\ldots(2n-3)\ ,$$

with a (shifted) *Catalan number* $\mathcal{C}_n = \frac{1}{n}\binom{2n-2}{n-1}$.

Now let us consider the average level of node j in a heap ordered tree of size n. For that purpose we need a different recursion than (1). We want to fix the subtree which contains j and say that it is the first subtree. However, j can be in any of the m subtrees. So we introduce a factor m. (Of course, we assume $j\ge2$.) But then, we cannot choose h_1 numbers out of $\{2,\ldots,n+1\}$, since we know already that j is there!

Therefore our alternative recursion is

$$a_{n+1} = \sum_{m\ge1}\sum_{h_1+\cdots+h_m=n} m\binom{n-1}{h_1-1,h_2,\ldots,h_m} a_{h_1}\ldots a_{h_m} \quad \text{for } n\ge1,\quad a_1=1\ . \tag{2}$$

Let us test via generating functions that this is indeed correct. We multiply (2) by $\frac{z^{n-1}}{(n-1)!}$ and sum up. The right hand side becomes

$$\frac{d}{du}\frac{u\,A'(z)}{1-u\,A(z)}\bigg|_{u=1} = \frac{1}{(1-2z)^{3/2}}\ .$$

The left hand side is $A''(z)$, which equals the same quantity.

Now we want the probability that j lies in a subtree of size k. This is

$$\sum_{m\ge1}\sum_{h_2+\cdots+h_m=n-k} m\binom{n-1}{k-1}\binom{n-k}{h_2,\ldots,h_m}\frac{a_k a_{h_2}\ldots a_{h_m}}{a_{n+1}}$$

We compute the factor of $\binom{n-1}{k-1}\frac{a_k}{a_{n+1}}$ in this expression by multiplying this factor by $\frac{z^{n-k}}{(n-k)!}$ and summing up. The result is amazingly simple:

$$\frac{d}{du}\frac{u}{1-u\,A(z)}\bigg|_{u=1} = \frac{1}{1-2z}\ ,$$

whence our sought probability that j lies in a subtree of size k turns out to be

$$\binom{n-1}{k-1}\frac{a_k}{a_{n+1}}\,2^{n-k}\,(n-k)!\ .$$

Again, it is an easy check that the sum on k of this quantity is 1.

Now, let $D_{n,j}$ be the average level of node j. To avoid misinterpretations, we say that the root (= the node '1') is on level 0. Using the last probability, we can set up a recursion for $D_{n,j}$. However, we must take care of the fact that in its subtree, 'j' does not mean 'j' any further. The numbers in the subtree are to be replaced by $1,2,\ldots,$ according to their relative order. Let us compute the probability that j will be i after this procedure, or, what is the same, that j is the ith largest number in its subtree. It is

$$\frac{\binom{j-2}{i-1}\binom{n+1-j}{k-i}}{\binom{n-1}{k-1}} ,$$

since $i-1$ numbers have to be chosen from $\{2,3,\dots,j-1\}$ and $k-i$ numbers from $\{j+1,\dots,n+1\}$.

Therefore we finally found our recursion:

$$D_{n+1,j} = 1 + \sum_{k=1}^{n} \binom{n-1}{k-1} \frac{a_k}{a_{n+1}} 2^{n-k} (n-k)! \sum_{i=1}^{k} \frac{\binom{j-2}{i-1}\binom{n+1-j}{k-i}}{\binom{n-1}{k-1}} D_{k,i} . \tag{3}$$

This recursion is valid for $n+1 \ge j \ge 2$. Additionally, we have, since '1' is always on level 0, that $D_{n,1} = 0$ for all $n \ge 1$.

The solution of this recursion is found by inspection[1] (!), and it transpires that

$$D_{n,j} = H_{2j-2} - \tfrac{1}{2} H_{j-1} , \quad \text{independently of } n \quad (n \ge j) .$$

Since we have such a simple result, we should *explain* it and *prove* it. The independency on n means that the nodes larger than j don't change anything. We might say that a random tree of n nodes (each heap ordered tree is equally likely) remains 'random' if we cut off the nodes larger than j. This is in fact not too hard to see. First, it is enough to show that randomness is preserved if we erase node n. Or, since

$$\frac{a_{n+1}}{a_n} = 2n - 1 ,$$

we might alternatively show that from each tree with n nodes, we get $2n-1$ new trees by inserting the new node $n+1$. We can attach this new node on every node, but the relative order in the plane is important, so if a certain node has i outgoing branches, it gives us $i+1$ possibilities, namely to the left of all, between first and second edge, etc. Denoting by $d(k)$ the number of outgoing branches, we have altogether

$$\sum_{k=1}^{n} \Big(1 + d(k)\Big) = n + \text{number of edges} = 2n - 1 ,$$

as desired.

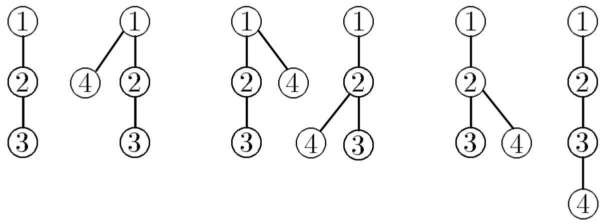

FIGURE 2. A heap ordered tree with 3 nodes and the 5 trees obtained by inserting a fourth node

[1]We used MAPLE to 'see' it.

To solve the recursion, we set $n = j-1$ and write $D_j = D_{j,j}$ for simplicity. Then (3) turns into

$$D_j = 1 + \sum_{k=1}^{j-1} \frac{a_k}{a_j}\, 2^{j-1-k}\,(j-1-k)!\binom{j-2}{k-1} D_k\ . \tag{4}$$

With

$$E_j := \frac{a_j D_j}{2^j (j-1)!}\ ,$$

we get

$$2(j-1)E_j = \frac{2^{1-j}a_j}{(j-2)!} + \sum_{k=1}^{j-1} E_k\ ,$$

or, by taking differences of two consecutive equations,

$$2jE_{j+1} = (2j-1)E_j + \frac{a_j}{2^j(j-1)!}\ ,$$

which means that

$$D_{j+1} = D_j + \frac{1}{2j-1}\ .$$

Unrolling this recursion, we get

$$D_j = 1 + \tfrac{1}{3} + \cdots + \tfrac{1}{2j-3} = H_{2j-2} - \tfrac{1}{2}H_{j-1}\ ,$$

and that was to be demonstrated.

We can get an easy corollary: Since

$$\sum_{j=1}^{n}\sum_{i=1}^{j-1} \frac{1}{2i-1} = \sum_{i=1}^{n} \frac{n-i}{2i-1} = \sum_{i=1}^{n} \frac{(n-\frac{1}{2})-(i-\frac{1}{2})}{2i-1} = \left(n-\frac{1}{2}\right)(H_{2n}-\tfrac{1}{2}H_n) - \frac{n}{2}\ ,$$

we get, by dividing this by n, the average depth d_n of a random node

$$d_n = \left(1-\frac{1}{2n}\right)(H_{2n}-\tfrac{1}{2}H_n) - \frac{1}{2}\ ,$$

a result from [9] and [2].

Now we want to compute the variance. Very much in the style of (3) we can set up a recursion for the probability generating functions $F_{n,j}(x)$;

$$F_{n+1,j}(x) = x\sum_{k=1}^{n}\binom{n-1}{k-1}\frac{a_k}{a_{n+1}}\, 2^{n-k}\,(n-k)!\sum_{i=1}^{k} \frac{\binom{j-2}{i-1}\binom{n+1-j}{k-i}}{\binom{n-1}{k-1}} F_{k,i}(x)\ .$$

As before, we freely drop the index n, by replacing n by $j-1$.

$$F_j(x) = x\sum_{k=1}^{j-1} \frac{a_k}{a_j}\, 2^{j-1-k}\,(j-1-k)!\binom{j-2}{k-1} F_k(x)\ . \tag{5}$$

With the short hand notation

$$E_j = \frac{a_j F_j(x)}{2^j(j-1)!}\ ,$$

we find

$$2(j-1)E_{j+1} = x\sum_{k=1}^{j-1} E_j \ ,$$

whence

$$2jE_{j+1} - 2(j-1)E_j = xE_j \ ,$$

or

$$E_j = \frac{2j-4+x}{2j-2}E_{j-1} = \frac{1}{2^{j-1}(j-1)!}\prod_{i=1}^{j-1}(2i-2+x) \ .$$

Therefore

$$F_j(x) = \prod_{i=1}^{j-1}\frac{2i-2+x}{2i-1} \ . \tag{6}$$

When we have the probability generating function in a factored form where each factor constitutes a probability generating function itself, it is particularly easy to get the variance, see [5].

$$V_j := \mathrm{Var}\left\{F_j(x)\right\} = \sum_{i=1}^{j-1}\mathrm{Var}\left\{\frac{2i-2+x}{2i-1}\right\} = \sum_{i=1}^{j-1}\left[\frac{1}{2i-1} - \frac{1}{(2i-1)^2}\right] \ .$$

The reader might have noticed that in this paper a modified notion of harmonic numbers (of first and second order) would be more appropriate. If we define

$$\widehat{H}_n = \sum_{k=1}^{n}\frac{1}{2k-1} \qquad \text{and} \qquad \widehat{H}_n^{(2)} = \sum_{k=1}^{n}\frac{1}{(2k-1)^2} \ ,$$

then we have

$$V_j = \widehat{H}_{j-1} - \widehat{H}_{j-1}^{(2)} \ ;$$

in terms of harmonic numbers this equals

$$V_j = H_{2j-2} - \tfrac{1}{2}H_{j-1} - H_{2j-2}^{(2)} + \tfrac{1}{4}H_{j-1}^{(2)} \ .$$

From (6) it is even possible to get an explicit expression for the coefficient of x^k, which is the probability that node j is at level k. We write

$$F_j(x) = \frac{x(x+2)\dots(x+2(j-1))}{a_j} = \frac{(-2)^{j-1}}{a_j}\left(-\frac{x}{2}\right)^{\underline{j-1}} = \frac{2^{j-1}}{a_j}\sum_{k=0}^{j-1}\left[{j-1 \atop k}\right]\frac{x^k}{2^k} \ ,$$

and therefore

$$[x^k]F_j(x) = \frac{2^{j-1}}{a_j}\left[{j-1 \atop k}\right]\frac{1}{2^k} \ .$$

Here, we used the notion of Stirling's cycle numbers $\left[{n \atop k}\right]$, compare [3].

For the readers convenience we collect our main findings as follows.

Theorem 1. *The average and the variance of the level (depth) of the node with label j in a heap ordered tree of at least j nodes is given by*

$$\begin{aligned} D_j &= \widehat{H}_{j-1} \ , \\ V_j &= \widehat{H}_{j-1} - \widehat{H}^{(2)}_{j-1} \ . \end{aligned}$$

The probability that node j has level k is given by

$$\frac{2^{j-1-k}}{a_j} \begin{bmatrix} j-1 \\ k \end{bmatrix} \ .$$

2. Binary trees

For the sake of completeness and comparison we briefly sketch the corresponding considerations for the instance of binary trees. However, *mutatis mutandis,* these are well known and old results, going back to Lynch, Hibbard, Knuth and others, see [4, 7, 6, 8].

The recursion for the increasing binary trees is

$$a_{n+1} = \sum_{k=0}^{n} \binom{n}{k} a_k a_{n-k}$$

with the obvious solution $a_n = n!$. The probability that j lies in a subtree of size k is

$$2\binom{n-1}{k-1}\frac{k!(n-k)!}{(n+1)!} = \frac{2k}{n(n+1)} \ .$$

The recursion for the probability generating functions is

$$F_{n+1,j}(x) = x \sum_{k=1}^{n} \frac{2k}{n(n+1)} \sum_{i=1}^{k} \frac{\binom{j-2}{i-1}\binom{n+1-j}{k-i}}{\binom{n-1}{k-1}} F_{k,i}(x) \ ,$$

and the specialization to $n = j-1$ is

$$F_j(x) = \frac{2x}{j(j-1)} \sum_{k=1}^{j-1} k\, F_k(x) \ ,$$

from which we derive

$$(j+1)F_{j+1}(x) = (j-1+2x)F_j(x)$$

and thus

$$F_j(x) = \prod_{i=2}^{j} \frac{i+2(x-1)}{i} \ .$$

Therefore the expectation is

$$D_j = \sum_{i=2}^{j} \frac{2}{i} = 2(H_j - 1) \ ,$$

and the variance is

$$V_j = \sum_{i=2}^{j} \left(\frac{2}{i} - \frac{4}{i^2} \right) = 2H_j - 4H_j^{(2)} + 2 \ .$$

Furthermore, the probability generating function $F_j(x)$ can be written as

$$F_j(x) = \frac{1}{j!}(-1)^{j-1}(-2x)^{\underline{j-1}} = \frac{1}{j!}\sum_{k=0}^{j-1}\begin{bmatrix} j-1 \\ k \end{bmatrix}(2x)^k ,$$

whence we find for the probability that the node j lies on level k the quantity

$$\frac{2^k}{j!}\begin{bmatrix} j-1 \\ k \end{bmatrix} .$$

References

[1] F. Bergeron, P. Flajolet, and B. Salvy. Varieties of increasing trees. *Lecture Notes in Computer Science*, 581:24–48, 1992.

[2] W.-C. Chen and W.-C. Ni. On the average altitude of heap–ordered trees. *International Journal of Foundations of Computer Science*, 15:99–109, 1994.

[3] R. L. Graham, D. E. Knuth, and O. Patashnik. *Concrete Mathematics*. Addison Wesley, 1989.

[4] T. N. Hibbard. Some combinatorial properties of certain trees with applications to searching and sorting. *Journal of the ACM*, 9(1):13–28, January 1962.

[5] D. E. Knuth. *The Art of Computer Programming*, volume 1: Fundamental Algorithms. Addison-Wesley, 1968. Second edition, 1973.

[6] D. E. Knuth. *The Art of Computer Programming*, volume 3: Sorting and Searching. Addison-Wesley, 1973.

[7] W. C. Lynch. More combinatorial problems on certain trees. *Computer Journal*, 7:299–302, 1965.

[8] H. M. Mahmoud. *Evolution of Random Search Trees*. John Wiley, New York, 1992.

[9] H. Prodinger. Depth and path length of heap ordered trees. *International Journal of Foundations of Computer Science*, 1996 (to appear).

[10] H. Prodinger and F.J. Urbanek. On monotone functions of tree structures. *Discrete Applied Mathematics*, 5:223–239, 1983.

Institut für Algebra und Diskrete Mathematik, Technical University of Vienna, Wiedner Hauptstrasse 8–10, A-1040 Vienna, Austria.

E-mail address: `Helmut.Prodinger@tuwien.ac.at`

WWW-address: `http://info.tuwien.ac.at/theoinf/proding.htm`