Forty years of tree enumeration

Helmut Prodinger

UNIVERSITEIT STELLENBOSCH UNIVERSITY

October 24, 2016

Canada

John Moon (Alberta)
Counting labelled trees
Canadian mathematical monographs

What is our subject?

Asymptotic enumeration
or
Probabilistic combinatorics?

What is our subject?

Asymptotic enumeration

AND

Probabilistic combinatorics!

What is our subject?

Exact enumeration!

Binary trees and exact enumeration

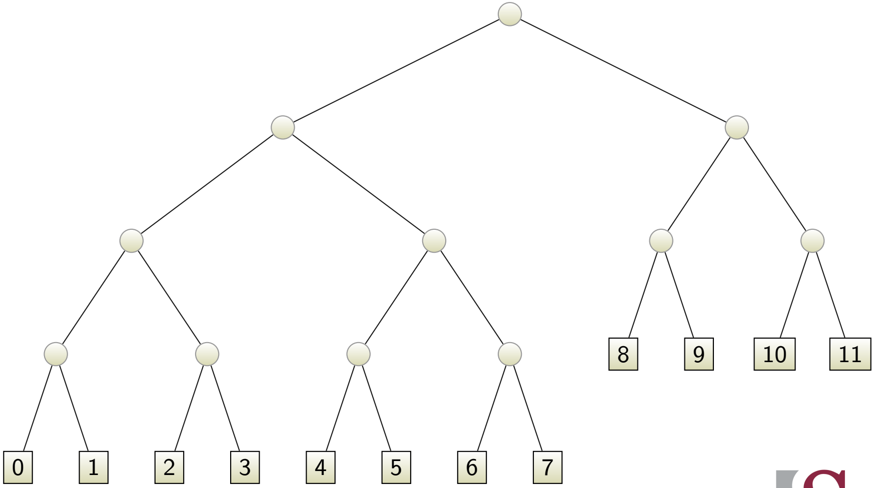


Figure: Leaves labelled left to right

Theorem

The expectation $E_{n,j}$ of the height of the leaf with label j in a binary tree of size n, is given by

$$E_{n,j} = \frac{2(2j+1)(2n-2j+1)}{n+2} \frac{\binom{2j}{j}\binom{2n-2j}{n-j}}{\binom{2n}{n}} \qquad \text{for} \quad 0 \le j \le n\,. \qquad \square \tag{1}$$

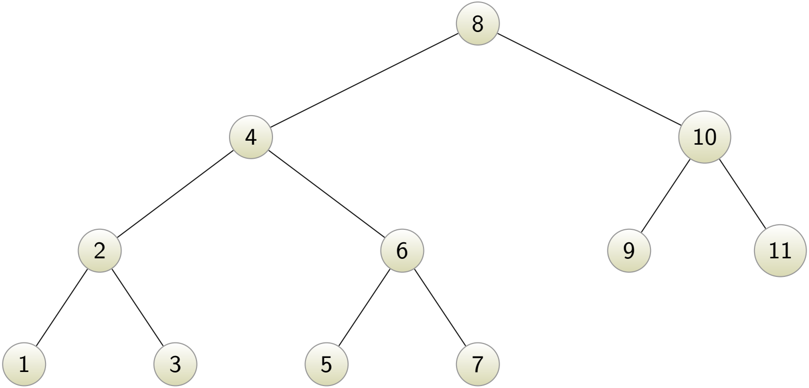


Figure: Nodes labelled via inorder traversal

Theorem

The expectation $E_{n,j}$ of the number of descendants of the node with label j, where the nodes are labelled by inorder traversal, in a binary tree of size n, is given by

$$E_{n,j} = \frac{n+1}{4} \frac{\binom{2j}{j}\binom{2(n+1-j)}{n+1-j}}{\binom{2n}{n}} \qquad \text{for } 1 \le j \le n$$

Binary trees and exact enumeration

Theorem

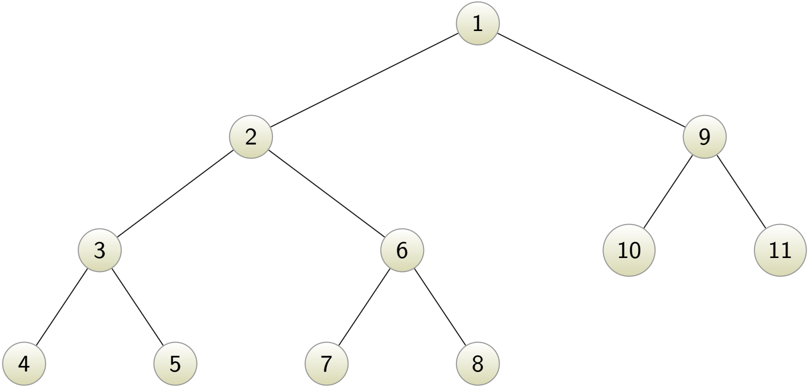
The expectation $E_{n,j}$ of the number of descendants of the node with label j, where the nodes are labelled by inorder traversal, in a binary tree of size n, is given by

$$E_{n,j} = \frac{n+1}{4} \frac{\binom{2j}{j}\binom{2(n+1-j)}{n+1-j}}{\binom{2n}{n}} \qquad \text{for } 1 \le j \le n$$

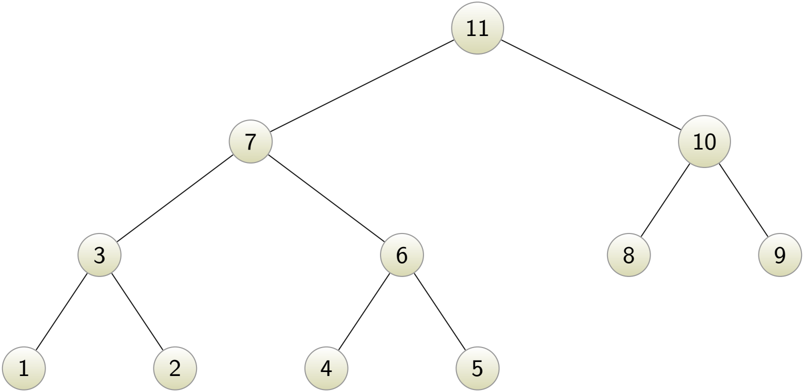
Many more results of this style are available (together with Alois Panholzer).

Trivariate generating functions and Zeilberger's algorithm.

Preorder



Postorder



Lattice paths also belong to tree enumeration!

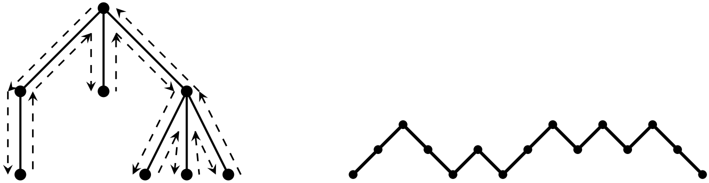


Figure: A planar tree with 8 nodes (=7 edges) and the corresponding Dyck path of length 14 (=semi-length 7)

Dyck like paths

Upsteps $+1$, downsteps -2
(in general $-(t-1)$)

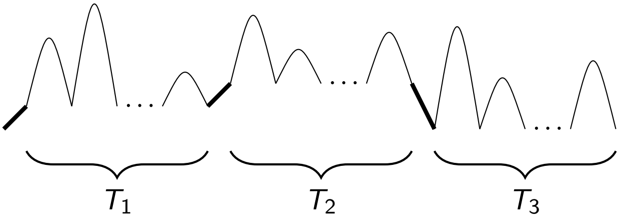


Figure: The decomposition of generalized Dyck paths leading (recursively) to a ternary tree with subtrees T_1, T_2, T_3.

Returns and hills

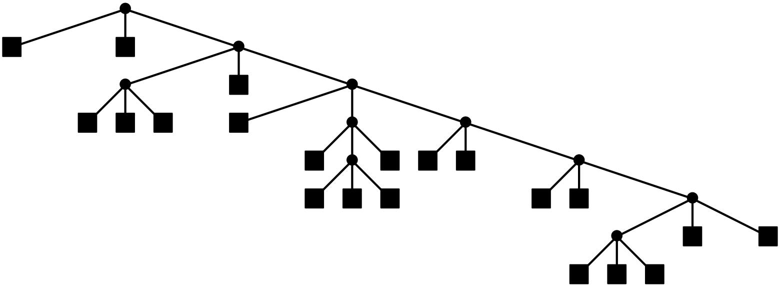
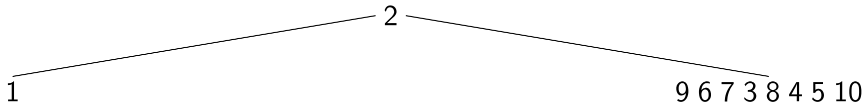


Figure: A ternary tree with 10 (internal) nodes. It has 6 returns and 3 hills.

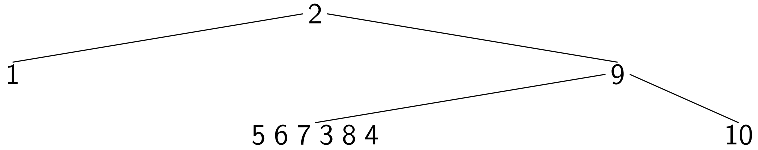
Explicit forms of bivariate generating functions are available, and results about the limiting distribution (negative binomial distribution)
Solving an open question.

Quickselect

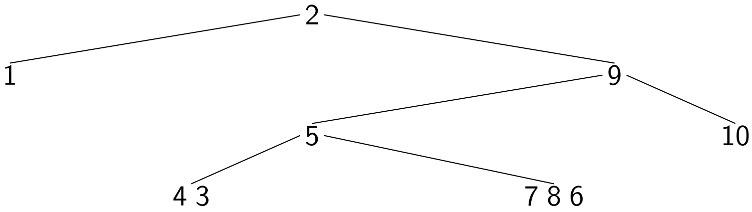
Permutation 2 1 9 6 7 3 8 4 5 10 Pivot is 2



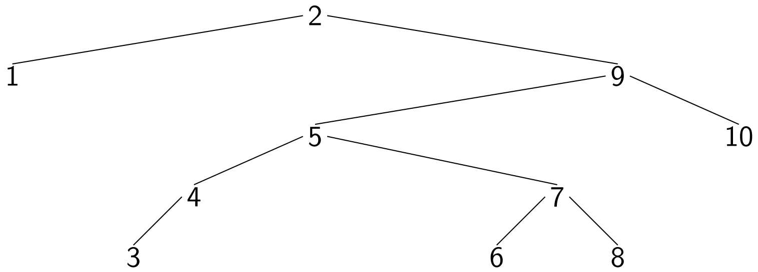
Quickselect



Quickselect



Quickselect



Quickselect

If one only wants the find the jth ranked element (which is j, since we assume that the elements in question are $\{1, 2, \dots, n\}$), one uses the same partitioning strategy as in Quicksort, but follows only the path which contains the sought element.
This is the same as to say that one goes down recursively in only *one* subfile. This procedure is called Hoare's FIND algorithm.
Knuth has already computed the average number of comparisons $C_{n,j}$. For this, it is assumed that every permutation of $\{1, 2, \dots, n\}$ is equally likely, and that the partitioning phase needs $n-1$ comparisons. Then there is the recursion

$$C_{n,j} = n - 1 + \frac{1}{n} \sum_{1 \le k < j} C_{n-k,j-k} + \frac{1}{n} \sum_{j < k \le n} C_{k-1,j} \ .$$

The solution is

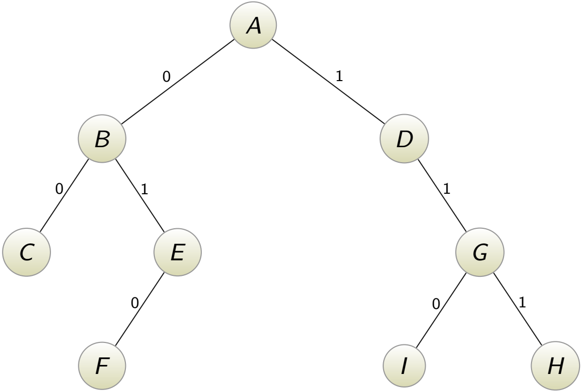
$$C_{n,j} = 2\Big(n + 3 + (n+1)H_n - (j+2)H_j - (n+3-j)H_{n+1-j}\Big) .$$

Quickselect: Second factorial moment (comparisons)

$$
\begin{aligned}
M_{n,j} &= -2(n+1)(n+6)\,H_n^2\\
&\quad - 8H_n\Big(jH_j + (n+1-j)\,H_{n+1-j}\Big)\\
&\quad + 4\,\frac{-(3n^2+8n+1)j^2 + (n+1)(3n^2+8n+1)j + 4(n+1)}{j(n+1-j)}\,H_n\\
&\quad + 2(j+8)(j+1)H_j^2 + 2(n+9-j)(n+2-j)\,H_{n+1-j}^2\\
&\quad + 4\Big(-j^2+(n+1)j-n^2-n+4\Big)H_j\,H_{n+1-j}\\
&\quad - \frac{2}{j(n+1-j)}\Big(-2(3n+7)j^3 + (6n^2+14n+13)j^2 + (n+1)(6n+1)j + 8(n+1)\Big)\,H_j\\
&\quad - \frac{2}{j(n+1-j)}\Big(2(3n+7)j^3 - (12n^2+46n+29)j^2\\
&\qquad\qquad + (n+1)(6n^2+26n+15)j + 8(n+1)\Big)\,H_{n+1-j}\\
&\quad + 2(n+1)(n+6)\,H_n^{(2)}\\
&\quad - 2(j^2+5j+8)\,H_j^{(2)} - 2(j^2-(2n+7)j+n^2+7n+14)\,H_{n+1-j}^{(2)}\\
&\quad + \frac{10j^4 - 20(n+1)j^3 + (n^2-13n-66)j^2 + (9n^2+33n+76)(n+1)j + 32}{2j(n+1-j)}\\
&\quad + 4n\Big((n+1-j)\sum_{k=1}^{j}\frac{H_{n-k}}{k} + j\sum_{k=1}^{n+1-j}\frac{H_{n-k}}{k}\Big)
\end{aligned}
$$

Digital search trees

A : 1001
B : 0110
C : 0000
D : 1111
E : 0100
F : 0101
G : 1101
H : 1110
I : 1100



Digital search trees: path length

Folklore: the average path length:

$$\sum_{k=2}^{n} \binom{n}{k} (-1)^k Q_{k-2}$$

with

$$Q_k = \left(1 - \frac{1}{2}\right)\left(1 - \frac{1}{4}\right) \dots \left(1 - \frac{1}{2^k}\right)$$

$$(x; q)_n = (1 - x)(1 - xq) \dots (1 - xq^{n-1})$$

Digital search trees

$$I_N = N - \sum_{k=2}^{N} \binom{N}{k} (-1)^k R_{k-2}$$

with

$$R_N = Q_N \sum_{k=0}^{N} \frac{1}{Q_k}$$

Asymptotic evaluation
P.Flajolet and R.Sedgewick, Digital Search Trees Revisited, SIAM J. Computing, 1986 748–767

Digital search trees: external internal nodes

External internal nodes in digital search trees via Mellin transforms.
H. Prodinger, SIAM Journal on Computing, 21:1180–1183, 1992.
Improved constant: nicer, fast convergent series:

$$\frac{N}{Q_\infty}\left(\frac{1}{\log 2}+\sum_{j\ge 2}(-1)^{j-1}2^{-\binom{j}{2}}/Q_{j-1}(j-1)\frac{1}{2^{j-1}-1}\right)$$

Final result

$$I_N \sim N\big(\alpha+1-R^*(-1)\big)=0.37204\cdots N.$$

A big challenge: Dealing with

$$\sum_{k=2}^{n-2} \binom{n}{k} Q_{k-2} Q_{n-k-2}$$

Digital search trees again revisited: The internal path length perspective. P. Kirschenhofer, H. Prodinger and W. Szpankowski, SIAM Journal on Computing, 23:598–616, 1994. (paper written 1987–1990)

Analysis of the variance of the path length. Complicated expressions!
Much improved a few years ago by Hwang, Fuchs, Zacharowitsch.
Approximating much earlier, much nicer constant.

Digital search trees: protected nodes

A : 1001
B : 0110
C : 0000
D : 1111
E : 0100
F : 0101
G : 1101
H : 1110
I : 1100

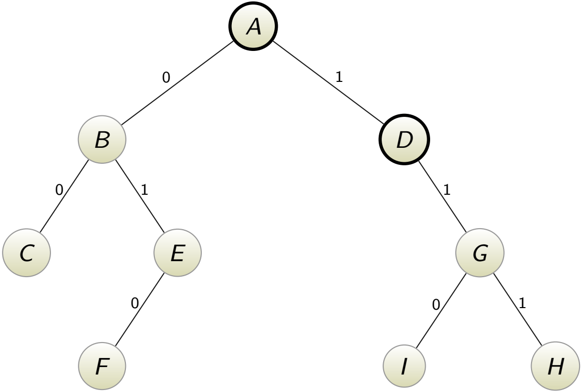


Figure: *A digital search tree with nine nodes, among which A and D are 2-protected.*

Digital search trees: protected nodes

Theorem

The average number of 2-protected nodes in random DSTs of size $N \ge 1$ is exactly given by

$$I_N = \sum_{k=2}^{N} \binom{N}{k} (-1)^k Q_{k-2} \sum_{n=1}^{k-2} \frac{1 - (n+1)2^{-n} - \frac{n(n+1)}{4}}{Q_n}.$$

Theorem

The average number I_N of 2-protected nodes in random DSTs of size N admits the asymptotic expansion

$$I_N = N \cdot 0.3070798139360592182854 9 \cdots + N \cdot \delta(\log_2 N) + O(1),$$

The tiny periodic function $\delta(x)$ has a Fourier expansion that could be computed in principle.

Rice's method

$$I_N = -\frac{1}{2\pi i}\int_{\mathcal{C}} \frac{\Gamma(N+1)\Gamma(-z)}{\Gamma(N+1-z)}\psi(z)dz,$$

where $\mathcal{C}$ encircles the poles $2, 3, \dots, N$ and no others. The function $\psi(z)$ is the extension of

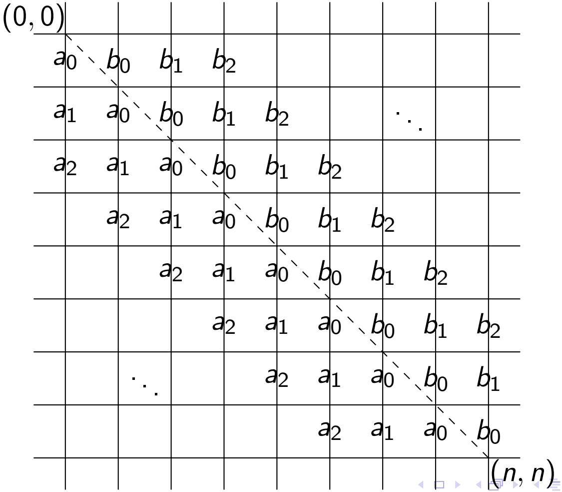
$$Q_{k-2}\sum_{n=1}^{k-2} \frac{1-(n+1)2^{-n}-\frac{n(n+1)}{4}}{Q_n}$$

Combinatorics of geometrically distributed random variables.

Combinatorics of geometrically distributed random variables.
Flajolet: “Prodinger’s *q*-analogs”

Batcher's odd-even exchange revisited

Batcher's odd-even exchange revisited: a generating functions approach , Helmut Prodinger, Theoretical Computer Science 636 (2016), 95–100.



Batcher

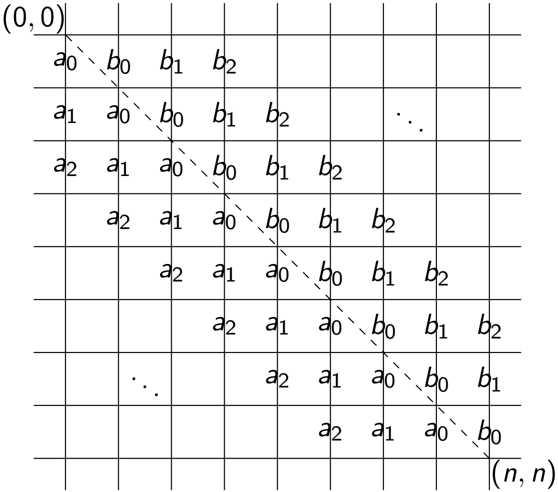
B_n: the average number of exchanges
Sedgewick:

$$B_n = \frac{1}{\binom{2n}{n}} \sum_{k \geq 1} \binom{2n}{n-k} (2F(k) + k), \tag{2}$$

where $F(k)$ is the summatory function of $f(j)$, which is the number of ones in the Gray code representation of j:

$$F(k) := \sum_{0 \leq j < k} f(j).$$

Batcher



The weights in our problem are $a_k = f(k)$ and $b_k = f(k) + 1$.

Batcher

New:
A generating function approach to derive Sedgewick's formula.

Batcher

New:

A generating function approach to derive Sedgewick's formula.

Method: Write symbolic equations for families of lattice paths, then translate them into generating functions

Batcher

A decorated path goes from $(0,0)$ to (n,n) and carries exactly one (vertical) label.
$\mathcal{W}$: the family of all paths $(0,0)$ to (n,n)
$\mathcal{D}$: $(0,0)$ to (n,n), staying on one (prescribed) side of the diagonal
$\mathcal{R}_p$: the family of paths with vertical label a_p,
$\mathcal{S}_p$: the family of paths with vertical label b_p.
We treat a_p as a *fixed* symbol (not depending on p).
Standard substition $z = \dfrac{u}{(1+u)^2}$.

Batcher

$$W(z) = \frac{1}{\sqrt{1-4z}} = \sum_{n\ge0} \binom{2n}{n} z^n = \frac{1+u}{1-u},$$

$$D(z) = \frac{1-\sqrt{1-4z}}{2z} = \sum_{n\ge0} \frac{1}{n+1}\binom{2n}{n} z^n = 1+u.$$

$$\mathcal{R}_p = \mathcal{W}\mathcal{A}_p\mathcal{W};$$

$$\mathcal{A}_p = \mathrm{d}\mathcal{D}\mathcal{A}_{p-1}\mathcal{D}\mathrm{h}, \quad p \ge 1, \quad \mathcal{A}_0 = a_p \cdot d\mathcal{D}h.$$

Batcher

$$A_p = z(1+u)^2 A_{p-1} = uA_{p-1}, \quad p \ge 1, \quad A_0 = a_p \frac{u}{1+u}.$$

By iteration, $A_p = a_p \dfrac{u^{p+1}}{1+u}$ and therefore

$$R_p = a_p \frac{u^{p+1}(1+u)}{(1-u)^2}$$

by symmetry

$$S_p = b_p \frac{u^{p+1}(1+u)}{(1-u)^2}.$$

Batcher

In the Batcher problem: $b_p = 1 + a_p$, $a_p = f(p)$
$f(k)$: number of ones in the Gray code representation of k.

$$B(z) := \sum_{p\ge0}(R_p + S_p) = \frac{u(1+u)}{(1-u)^3} + 2\sum_{p\ge0} f(p)\frac{u^{p+1}(1+u)}{(1-u)^2}.$$

$$B(z) = \frac{u(1+u)}{(1-u)^3} + 2\frac{u(1+u)}{(1-u)^3}\sum_{k\ge0}\frac{u^{2^k}}{1+u^{2^{k+1}}}.$$

Formula (with small mistake) given earlier by Knuth (without proof).
Sedgewick's formula follows from this by standard extraction of coefficients.

Batcher

Starting from

$$B(z) = \frac{u(1+u)}{(1-u)^3} + 2\frac{u(1+u)}{(1-u)^3}\sum_{k\ge0}\frac{u^{2^k}}{1+u^{2^{k+1}}}.$$

one can also do asymptotics.
Singularity of generating functions, Mellin transform.

Theorem

The average number of exchanges in the odd-even merge of $2n$ *elements satisfies*

$$B_n \sim \frac{1}{4} n \log_2 n + nB(\log_4 n),$$

where $B(x)$ *is a continuous periodic function of period 1; this function can be expanded as a Fourier series* $B(x) = \sum_{k\in\mathbb{Z}} b_k e^{2k\pi ix}$, *with*

$$b_0 = -\frac{1}{2\log 2} - \frac{\gamma}{4\log 2} - \frac{3}{4} + 2\log_2 \Gamma\left(\tfrac{1}{4}\right) - \log_2 \pi \approx 0.385417224$$

and for $k \neq 0$, *with the abbreviation* $\chi_k = \dfrac{2\pi ik}{\log 2}$,

$b_k = \dfrac{1}{\log 2}\zeta\left(\chi_k, \tfrac{1}{4}\right)\dfrac{\Gamma(\chi_k/2)}{1+\chi_k}$. *Furthermore,* $|B(x) - b_0| \leq 0.0005$.

Asymptotics in the early days

De Bruijn's book Asymptotic methods in Analysis

De Bruijn's book Asymptotic methods in Analysis
One chapter in Comtet's book Advanced Combinatorics
(Darboux's lemma)

Asymptotics in the early days

De Bruijn's book Asymptotic methods in Analysis
One chapter in Comtet's book Advanced Combinatorics (Darboux's lemma)
Edward A. Bender's survey article, SIAM Rev., 16 (1974), 485–515. (31 pages) Asymptotic Methods in Enumeration

Asymptotics in the early days

De Bruijn's book Asymptotic methods in Analysis
One chapter in Comtet's book Advanced Combinatorics (Darboux's lemma)
Edward A. Bender's survey article, SIAM Rev., 16 (1974), 485–515. (31 pages) Asymptotic Methods in Enumeration
much later Odlyzko's survey paper.

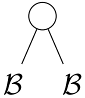
Asymptotics in the early days

De Bruijn's book Asymptotic methods in Analysis
One chapter in Comtet's book Advanced Combinatorics (Darboux's lemma)
Edward A. Bender's survey article, SIAM Rev., 16 (1974), 485–515. (31 pages) Asymptotic Methods in Enumeration
much later Odlyzko's survey paper.
Now: The book Analytic Combinatorics and various survey papers by Philippe Flajolet (1948–2011)

Register function

Work with Benjamin Hackl and Clemens Heuberger, in progress.
Binary trees are either a leave or a root together with a left and a right subtree, which are binary trees.
Symbolic equation:

$$\mathcal{B} = \square +$$



A binary tree of size n has n internal nodes, and thus $n+1$ external nodes (leaves). The number b_n of binary trees of size n is the nth Catalan number

$$C_n = \frac{1}{n+1}\binom{2n}{n},$$

which follows the generating function

$$B(z) = \sum_{n\geq 0} b_n z^n = 1 + B^2(z) = \frac{1-\sqrt{1-4z}}{2z} = \sum_{n\geq 0} \frac{1}{n+1}\binom{2n}{n} z^n.$$

The equation

$$B(z) = 1 + \frac{z}{1-2z} B\Big(\frac{z^2}{(1-2z)^2}\Big).$$

(Touchard's identity) can also be seen as a recursive process to generate binary trees via

$$B_0(z) = 1, \quad B_r(z) = 1 + \frac{z}{1-2z} B_{r-1}\Big(\frac{z^2}{(1-2z)^2}\Big), \quad r \ge 1.$$

In this way we get

$$B_1(z) = 1 + z + 2z^2 + 4z^3 + 8z^4 + 16z^5 + 32z^6 + 64z^7 + 128z^8 + 256z^9 + 512z^{10} + \cdots,$$
$$B_2(z) = 1 + z + 2z^2 + 5z^3 + 14z^4 + 42z^5 + 132z^6 + 428z^7 + 1416z^8 + 4744z^9 + \cdots,$$
$$B_3(z) = 1 + z + 2z^2 + 5z^3 + 14z^4 + 42z^5 + 132z^6 + 429z^7 + 1430z^8 + 4862z^9 + \cdots.$$

The register function is recursively defined:
Recursive description : $\text{reg}(\Box) = 0$, and if tree t has subtrees t_1 and t_2, then

$$\text{reg}(t) = \begin{cases} \max\{\text{reg}(t_1), \text{reg}(t_2)\} & \text{if } \text{reg}(t_1) \neq \text{reg}(t_2), \\ 1 + \text{reg}(t_1) & \text{otherwise.} \end{cases}$$

Reduction of binary trees; Register function

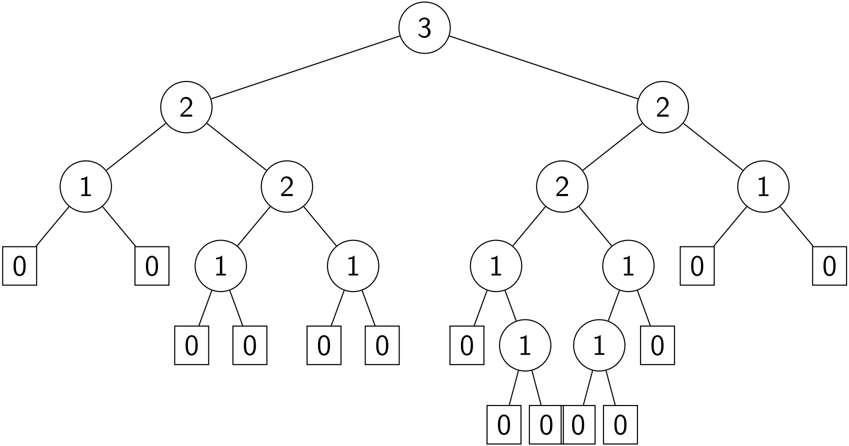


Figure: A binary tree with 13 internal nodes. The numbers in the nodes are the register function of the subtree having this node as root. The register function of the tree is the value at the root, i. e., 3.

Reduction of binary trees; Register function

Classical results (Flajolet et al.; Kemp, 1979)
The average value of the register function, assuming that all binary trees of size n ($= n$ internal nodes), is asymptotically given as

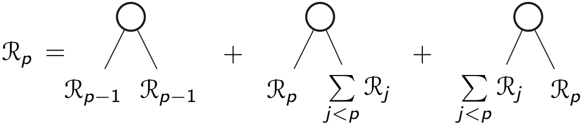
$$\log_4 n + \delta(\log_4 n)$$

with a periodic function $\delta(x)$.

The register function is also known as Horton-Strahler numbers in the study of the complexity of river networks.

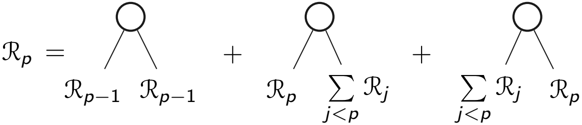
Reduction of binary trees; Register function

Let $\mathcal{R}_p$ denote the family of trees with register function $= p$, then one gets immediately from the recursive definition:



Reduction of binary trees; Register function

Let $\mathcal{R}_p$ denote the family of trees with register function $= p$, then one gets immediately from the recursive definition:

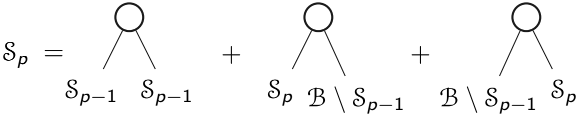


$$R_p = zR_{p-1}^2 + 2R_p \sum_{j<p} R_j$$

Reduction of binary trees; Register function

Easier to manipulate:

Easier to manipulate:
Let $\mathcal{S}_p$ denote the family of trees with register function $\geq p$, then one gets immediately from the recursive definition:



Reduction of binary trees; Register function

$$S_p(z) = \frac{1-u^2}{u}\frac{u^{2^p}}{1-u^{2^p}}.$$

$$R_p(z) = \frac{1-u^2}{u}\frac{u^{2^p}}{1-u^{2^{p+1}}}.$$

$$z = \frac{u}{(1+u)^2}$$

Reduction of binary trees; Register function

Flajolet's approach: relatively elementary, using the dyadic valuation $v_2(n)$:
If $n = 2^i(2j+1)$, then $i = v_2(n)$.
Can be linked to the sum of digits function $S_2(n)$.
A result by H. Delange on the average value of the sum of digits function can be used.

Kemp's approach: Mellin transform.
At that period, people called it the "Gamma function method".

Reduction of binary trees; Register function

Today: Recursion at the bottom (at the leaves).

Reduction of binary trees; Register function

Today: Recursion at the bottom (at the leaves).
Compactification of binary trees, which we write as $\Phi(t)$:

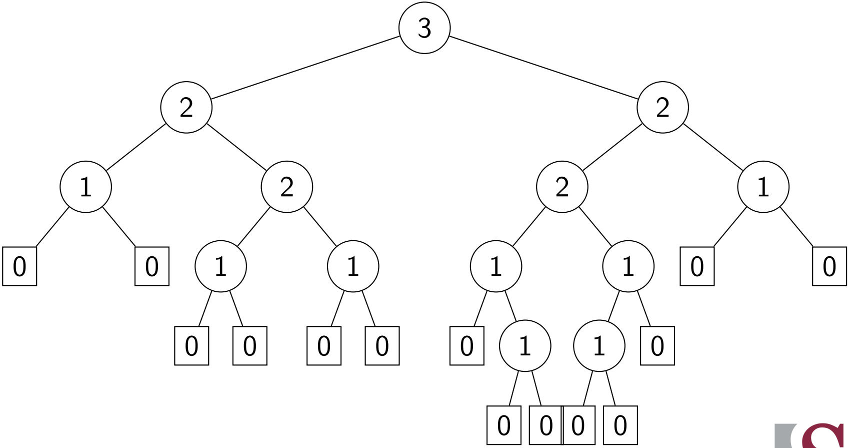
Reduction of binary trees; Register function

Today: Recursion at the bottom (at the leaves).
Compactification of binary trees, which we write as $\Phi(t)$:
The leaves (external nodes) will be erased.

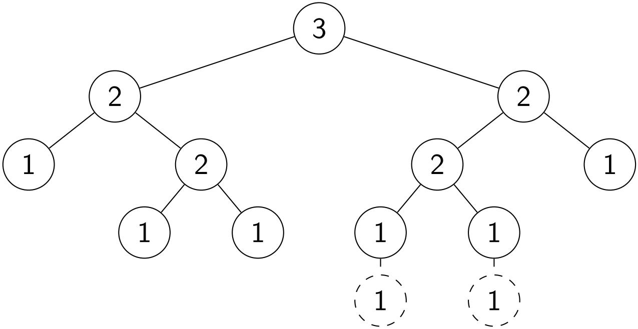
Reduction of binary trees; Register function

Today: Recursion at the bottom (at the leaves).
Compactification of binary trees, which we write as $\Phi(t)$:
The leaves (external nodes) will be erased.
Then, if a node has only one off-spring, these two nodes will be merged; this operation will be repeated as long as there are such nodes. Finally, the endnodes are declared to be external nodes.
This operation was introduced by two japanese physicists.

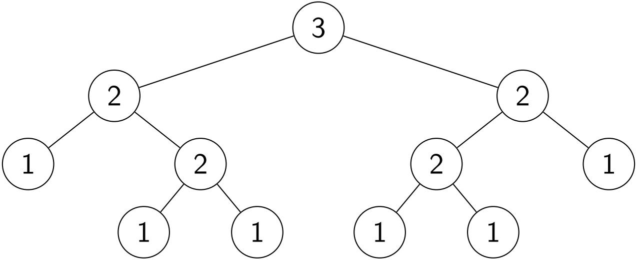
Reduction of binary trees; Register function



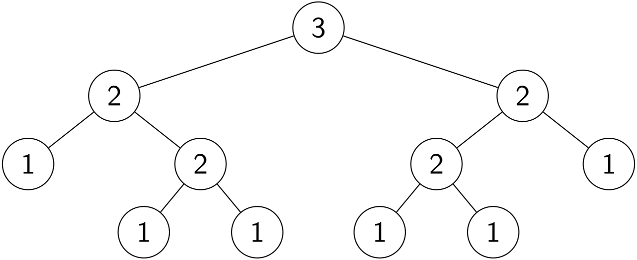
Reduction of binary trees; Register function

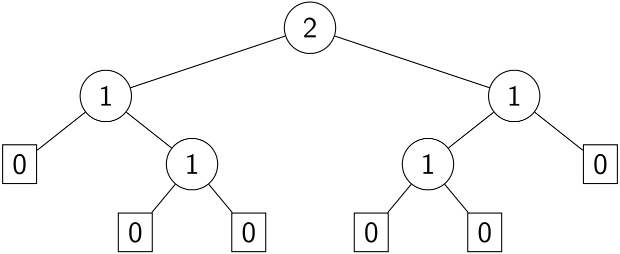


Reduction of binary trees; Register function



Reduction of binary trees; Register function





Reduction of binary trees; Register function

Note that $\Phi(\square)$ is undefined, so this is a partial function. Of course, many different trees are mapped to the same binary tree. However, they can all be obtained from a given reduced tree by the following operations:

Note that $\Phi(\Box)$ is undefined, so this is a partial function. Of course, many different trees are mapped to the same binary tree. However, they can all be obtained from a given reduced tree by the following operations:
Each leaf can be replaced by an internal node and an arbitrary chain of internal nodes on top, where the branches may be left or right ones. Thus, if the leaf is replaced by a chain of $k \geq 1$ internal nodes, this leads to 2^{k-1} choices. Similarly, an internal node is replaced by an internal and an arbitrary chain of internal nodes on top, where the branches may left or right ones. Eventually, the resulting tree is completed by external nodes in the usual way.

If $F(z)$ is a generating function counting some binary trees, then $vF(zv)$ counts them with respect to size (variable z) and number of leaves (variable v).

Reduction of binary trees; Register function

If $F(z)$ is a generating function counting some binary trees, then $vF(zv)$ counts them with respect to size (variable z) and number of leaves (variable v). The substitution process just described means that $v \mapsto \frac{z}{1-2z}$ and $z \mapsto \frac{z}{1-2z}$.

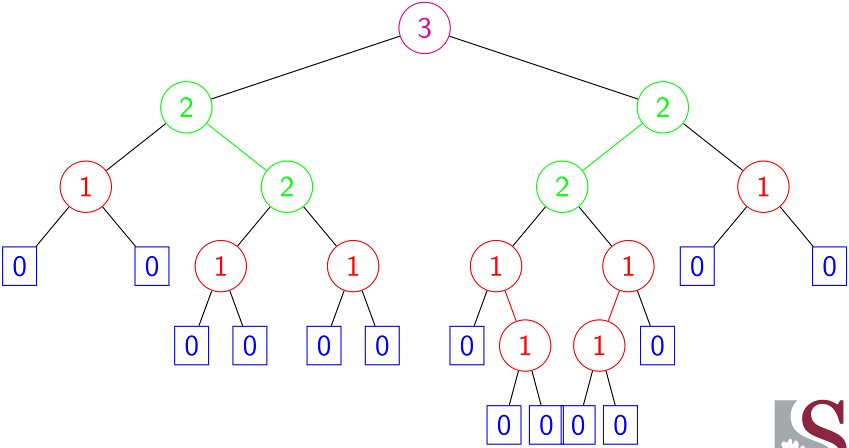
If $F(z)$ is a generating function counting some binary trees, then $vF(zv)$ counts them with respect to size (variable z) and number of leaves (variable v). The substitution process just described means that $v \mapsto \frac{z}{1-2z}$ and $z \mapsto \frac{z}{1-2z}$. Altogether, this results in

$$\frac{z}{1-2z} F\Big(\frac{z^2}{(1-2z)^2}\Big).$$

Reduction of binary trees; Register function

We have

$$\Phi^r(t) \in \mathcal{B} \Leftrightarrow \mathrm{Reg}(t) \geq r.$$



Thus, if we set

$$F^{(r)}(z) = \sum_{t:\,\mathrm{Reg}(t)\ge r} z^{|t|},$$

we get

$$F^{(0)}(z) = B(z), \quad F^{(r)}(z) = \frac{z}{1-2z}F^{(r-1)}\Big(\frac{z^2}{(1-2z)^2}\Big), \quad r \ge 1.$$

The substitution $z = \frac{u}{(1+u)^2}$ is always a good idea when dealing with the register function or Catalan numbers in general. Then, $\sigma(z) := \frac{z^2}{(1-2z)^2} = \frac{u^2}{(1+u^2)^2}$, so it just means $u \mapsto u^2$. Furthermore, $\frac{z}{1-2z} = \frac{u}{1+u^2}$. Note also that $F^{(0)}(z) = B(z) = 1 + u$.

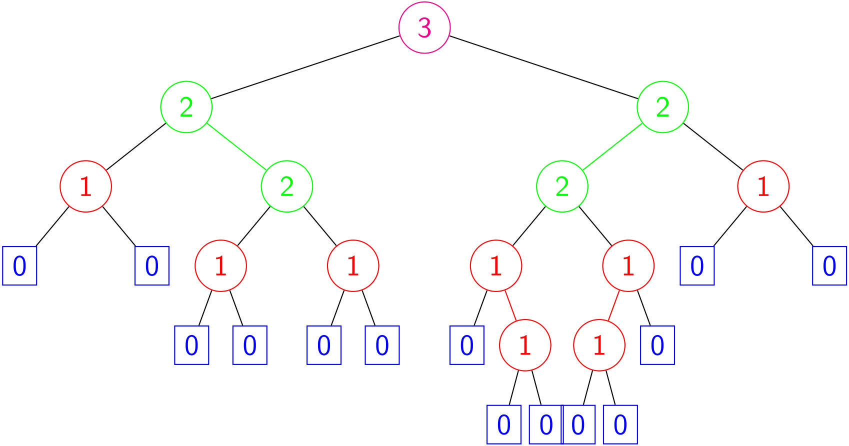
$$F^{(r)}(z) = \frac{u}{1+u^2}\frac{u^2}{1+u^4}\cdots\frac{u^{2^{r-1}}}{1+u^{2^r}}F^0(\sigma^r(z))$$
$$= \frac{1-u^2}{u}\frac{u^{2^r}}{1-u^{2^{r+1}}}(1+u^{2^r}) = \frac{1-u^2}{u}\frac{u^{2^r}}{1-u^{2^r}}.$$

This formula for the generating function of the number of trees with register function $\geq r$ is of course well known.
Likewise, the generating function $B_r(z)$ has the number of trees of size n and register function $\leq r$ as coefficients.

$$B_0(z) = 1, \quad B_r(z) = 1 + \frac{z}{1-2z} B_{r-1}\Big(\frac{z^2}{(1-2z)^2}\Big), \quad r \geq 1.$$

r-branches

Reduction of binary trees; Register function



An r-branch is a maximal chain of nodes labelled r. This must be a chain, since the merging of two such chains would already result in the higher value $r+1$. The nodes of the tree are partitioned

Parameter “number of r-branches”, in particular, the average number of them, assuming that all binary trees of size n are equally likely.

Parameter "number of r-branches", in particular, the average number of them, assuming that all binary trees of size n are equally likely.

Explicit formula for the expectation (and, in principle, also for higher moments).

Parameter "number of r-branches", in particular, the average number of them, assuming that all binary trees of size n are equally likely.
Explicit formula for the expectation (and, in principle, also for higher moments).
Total number of r-branches, for any r, i.e., the sum over $r \geq 0$.

Reduction of binary trees; Register function

The r-branches are 0-branches after r iterations of Φ. The 0-branches are just the leaves; they are the only nodes labelled 0, and they form a branch for itself. So, we have again the generating function $vB(zv)$. We start by computing average values. Then we have to compute

$$\frac{\partial}{\partial v} vB(zv)\Big|_{v=1} = \frac{1}{\sqrt{1-4z}} = \frac{1+u}{1-u}.$$

Again we have the recursion

$$G^{(0)}(z) = \frac{1}{\sqrt{1-4z}}, \quad G^{(r)}(z) = \frac{z}{1-2z} G^{(r-1)}\Big(\frac{z^2}{(1-2z)^2}\Big), \quad r \ge 1,$$

this time for $G^{(r)}(z)$. Note that

$$E_{n;r} := \frac{1}{C_n}[z^n]G^{(r)}(z)$$

is the average number of r-branches in a random tree of size n.

Reduction of binary trees; Register function

Iteration leads now to

$$G^{(r)}(z) = \frac{1-u^2}{u}\frac{u^{2^r}}{1-u^{2^{r+1}}} \cdot \frac{1+u}{1-u}\Big|_{u\mapsto u^{2^r}} = \frac{1-u^2}{u}\frac{u^{2^r}}{(1-u^{2^r})^2}.$$

Expanding this function about $u=1$ means expanding it in terms of $\sqrt{1-4z}$. This can be done with a computer

$$G^{(r)}(z) \sim \frac{1}{4^r\sqrt{1-4z}} + \frac{1}{3}(4^{-r}-1)\sqrt{1-4z} + \frac{1}{15}(4^{1-r}-5+4^r)(1-4z)^{3/2} + \cdots .$$

Singularity analysis guarantees that one can read off coefficients in this expansion:

$$[z^n]G^{(r)}(z) \sim \frac{1}{4^r}4^n\binom{-\frac{1}{2}}{n}(-1)^n + \frac{1}{3}(4^{-r}-1)4^n\binom{\frac{1}{2}}{n}(-1)^n + \frac{1}{15}(4^{1-r}-5+4^r)4^n\binom{\frac{3}{2}}{n}(-1)^n + \cdots$$

$$\sim \frac{4^n}{\sqrt{\pi}}\left(\frac{1}{4^r\sqrt{n}} + \frac{1}{6}\left(1-\frac{7}{4^{r+1}}\right)\frac{1}{n^{3/2}}\right) + \cdots$$

Reduction of binary trees; Register function

The asymptotics of C_n are straight forward, especially for a computer, and eventually we find

$$\frac{1}{C_n}[z^n]G^{(r)}(z) \sim \frac{n}{4^r} + \frac{1}{6}\Big(\frac{5}{4^r}+1\Big) + \frac{1}{20n}\Big(4^r - \frac{1}{4^r}\Big) + \cdots .$$

In principle, any number of terms would be available.
Variance can also be computed.

Reduction of binary trees; Register function

Theorem

The number of r-branches in binary trees of size n has for expectation and variance the following asymptotic formulæ, which hold for fixed r and $n \to \infty$:

$$E_{n;r} = \frac{n}{4^r} + \frac{1}{6}\Big(\frac{5}{4^r} + 1\Big) + \frac{1}{20n}\Big(4^r - \frac{1}{4^r}\Big) + O\Big(\frac{1}{n^2}\Big),$$

$$V_{n;r} = \frac{4^r - 1}{3\,16^r}n - \frac{23}{90}16^{-r} + \frac{5}{18}4^{-r} - \frac{1}{45} + O\Big(\frac{1}{n}\Big).$$

Reduction of binary trees; Register function

Theorem

The expected number of r-branches in binary trees of size n is given by the explicit formula

$$\frac{n+1}{\binom{2n}{n}}\sum_{\lambda\ge1}\lambda\left[\binom{2n}{n+1-\lambda 2^r}-2\binom{2n}{n-\lambda 2^r}+\binom{2n}{n-1-\lambda 2^r}\right].$$

Reduction of binary trees; Register function

Theorem

The average value of the total number of branches in a random binary tree of size n admits the asymptotic expansion

$$\frac{4n}{3} + \frac{1}{12}\log_2 n - \frac{2\zeta'(-1)}{\log 2} - \frac{\gamma}{6\log 2} + \frac{37}{36} + \delta(\log_4 n) + o(1),$$

with

$$\delta(x) = \frac{1}{\log 2}\sum_{k\neq 0}\Gamma(\frac{\chi_k}{2})\zeta(\chi_k - 1)(\chi_k - 1)e^{2\pi ikx}.$$

The periodic function $\delta(x)$ is given by its Fourier series expansion; such functions are typical in a register context.

A Similar Recursive Scheme Involving Lattice Paths

Simple two-dimensional lattice paths are sequences of the symbols $\{\uparrow, \rightarrow, \downarrow, \leftarrow\}$. It is easy to see that the generating function counting these paths (without the path of length 0) is

$$L(z) = \frac{4z}{1-4z} = 4z + 16z^2 + 64z^3 + 256z^4 + 1024z^5 + \cdots.$$

Theorem

The generating function $L(z) = \frac{4z}{1-4z}$ satisfies the functional equation

$$L(z) = 4L\Big(\frac{z^2}{(1-2z)^2}\Big) + 4z.$$

Study of the "compactification degree."

Reduction of planar trees (ongoing research together with Benjamin Hackl and Sara Kropf)

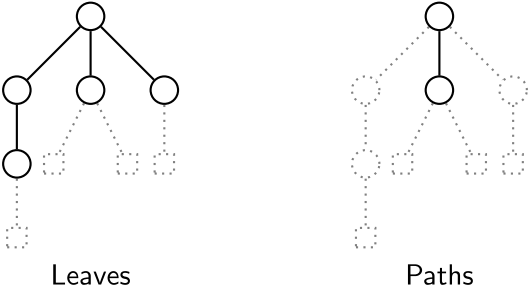


Figure: Removal of leaves / paths.

Reduction of planar trees

Proposition

The generating function $T(z,t)$ which enumerates rooted plane trees with respect to their internal nodes (marked by the variable z) and leaves (marked by t) is given explicitly by

$$T(z,t) = \frac{1-(z-t)-\sqrt{1-2(z+t)+(z-t)^2}}{2}.$$

Definition

The *Narayana numbers* are defined as

$$N_{n,k} = \frac{1}{n}\binom{n}{k-1}\binom{n}{k}$$

The *Narayana polynomials* are defined as

$$N_n(x) = \sum^{n} N_{n,k} x^{k-1}$$

Reduction of planar trees

The operator that drives the reduction of leaves:

$$\Phi(f(z,t)) := (1-t)f\left(\frac{z}{(1-t)^2}, \frac{zt}{(1-t)^2}\right).$$

Theorem

The bivariate generating function $G_r(z,v)$ enumerating rooted plane trees whose leaves can be cut at least r-times, where z marks the tree size and v marks the size of the r-fold cut tree, is given by

$$G_r(z,v) = \Phi^r T(zv, tv)\big|_{t=z}$$

and, equivalently, by

$$G_r(z,v) = \frac{1-u^{r+2}}{(1-u^{r+1})(1+u)} T\left(\frac{u(1-u^{r+1})^2}{(1-u^{r+2})^2} v, \frac{u^{r+1}(1-u)^2}{(1-u^{r+2})^2} v\right),$$

where $z = u/(1+u)^2$

Many explicit results can be deduced from that. E.g.

$$\begin{aligned}\mathbb{E}X^{d}_{n,r} &= \frac{1}{C_{n-1}}[z^n]\frac{\partial^d}{\partial v^d}G_r(z,1)\Big|_{v=1}\\ &= \frac{1}{C_{n-1}}[z^n]\frac{u^d d!}{(1+u)(1-u^{r+1})^d(1-u)^{d-1}}\tilde{N}_{d-1}(u^r)\end{aligned}$$

Priority trees

Average Case-Analysis of Priority trees: A structure for priority queue administration A. Panholzer and H. Prodinger, Algorithmica 22 (1998), 600–630.

Priority trees

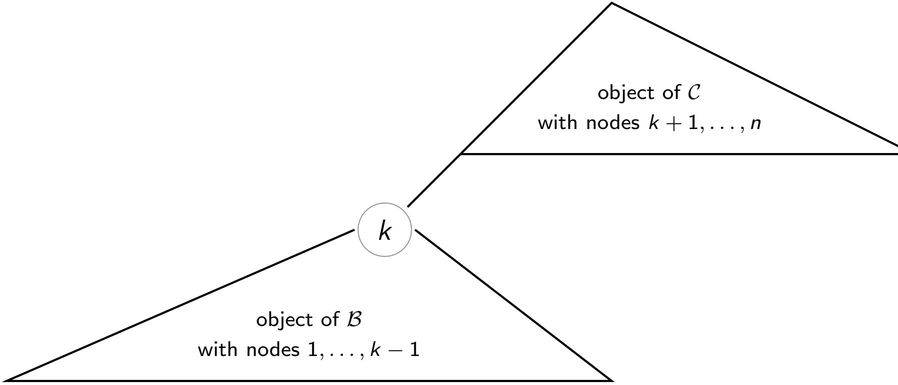


Figure: *Decomposition of the family $\mathcal{A}$.*

Priority trees

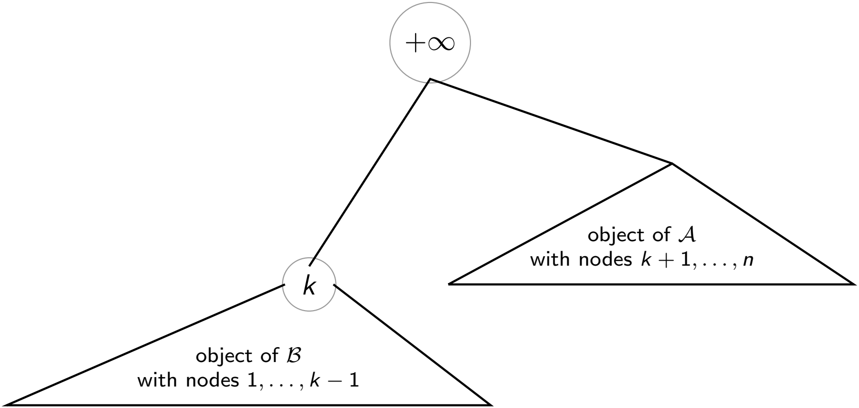


Figure: *Decomposition of the family $\mathcal{B}$.*

Priority trees

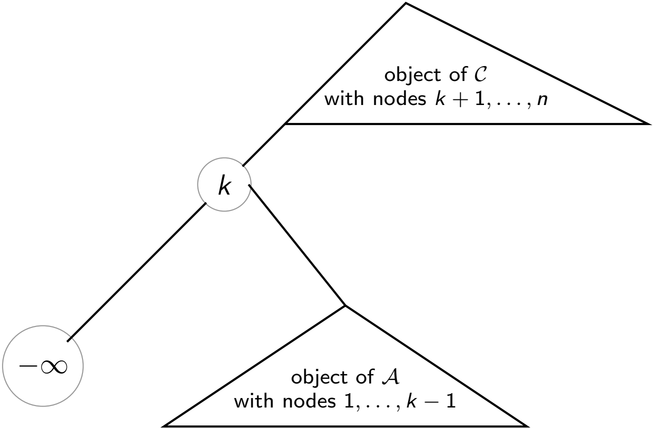


Figure: *Decomposition of the family $\mathcal{C}$.*

Tree function

The tree function

$$y = ze^y$$

enumerates the labelled rooted trees:

$$y(z) = \sum_{n\geq 1} n^{n-1}\frac{z^n}{n!}$$

a variant of Lambert's W-function

Tree function

Epidemics with two levels of mixing: The exact moments, H. Prodinger, SADIO 2 (1999), 1–4.
The probabilities

$$\binom{n-1}{k-1} p(pk)^{k-2}(1-pk)^{n-k}, \qquad (1 \le k \le n)$$

where considered in the study of an epidemics model.

$$\mathcal{E}(z) = \sum_{k \ge 0} (tk+1)^{k-1} \frac{z^k}{k!},$$

which is also given implicitly by

$$z = \mathcal{E}^{-t} \log \mathcal{E}.$$

A variant of the tree function.

Tree function

An identity conjectured by Lacasse via the tree function, H. Prodinger, Electronic Journal of Combinatorics 20 (3), 2013, P7.

$$\alpha(n)=\sum_{k=0}^{n}\binom{n}{k}k^k(n-k)^{n-k}$$

$$\beta(n)=\sum_{k_1+k_2+k_3=n}\frac{n!}{k_1!k_3!k_3!}k_1^{k_1}k_2^{k_2}k_3^{k_3}$$

$$\beta(n)-\alpha(n)=n^{n+1}$$

$$\alpha(n)=n![z^n]\Big(\frac{1}{1-y}\Big)^2 \quad \beta(n)=n![z^n]\Big(\frac{1}{1-y}\Big)^3$$

Tight Bounds on Information Leakage from Repeated Independent Runs
by Smith and Smith (2016)

Donald E. Knuth and Boris Pittel. A recurrence related to trees. Proceedings of the American Mathematical Society, 105(2):335—349.

Tree function

On Ramanujan's Q(n)-function. P. Flajolet, P. Grabner, P. Kirschenhofer and H. Prodinger, Journal of Computational and Applied Mathematics, 58:103–116, 1995.

$$Q(n) = 1 + \frac{n-1}{n} + \frac{(n-1)(n-2)}{n^2} + \cdots$$

$$R(n) = 1 + \frac{n}{n+1} + \frac{n^2}{(n+1)(n+2)} + \cdots$$

"Show that

$$R(n) - Q(n) = \frac{2}{3} + \frac{8}{135(n+k)}$$

where $k \equiv k(n)$ lies between $\frac{2}{21}$ and $\frac{8}{45}$"

Tree function

$$D(n) = R(n) - Q(n)$$

$$\sum_{n=1}^{\infty} D(n) n^{n-1} \frac{z^n}{n!} = \log \frac{(1-y)^2}{2(1-ez)} = \log \frac{(1-y)^2}{2(1-ye^{1-y})}$$

$$y = ze^y$$