Structured Numbers

Vincent Blondel
INRIA Rocquencourt

April 10, 1995

[summary by Philippe Flajolet]

Abstract

The talk describes a "lifting" of the system of unary representations of numbers into a system of tree-like representations. Alternatively, this can be seen as an arithmetic description of certain combinatorial properties of trees. In particular, addition, multiplication, and exponentiation of trees can be defined in a natural way.

This talk is based on [1]. We start with the family of *complete binary trees* [3], where each node has either 0 or 2 successors. The trees considered are rooted and embedded in the plane, so that left and right are distinguished. Nodes without successors are the external nodes, sometimes called leaves. The *weight* or size of a tree t is taken to be the number of its external nodes and is denoted by $|t|$. It has been well-known for over a century (bracketing problems, see [2]) that the number of trees of size n is given by the Catalan number,

$$T_n = \frac{1}{n}\binom{2n-2}{n-1}. \tag{1}$$

One may well view a tree of size n as a tree-like representation of integer n, and try to generalize the usual operations of addition, multiplication, and so on, of the integers. In other words, we also regard trees as extending the unary representation of integers with some supplementary structure superimposed, hence the name of "*structured numbers*" in the title.

1. Operations

First, *addition* is the basic operation defined as associating to two trees, u and v, the tree

$$u + v := (u \cdot v)$$

obtained by taking a root node and appending u and v to it, as left and right subtrees respectively. Given that weight is defined by number of external nodes, one has $|t| = |u| + |v|$, so we do capture in a way the usual addition of integers. On the other hand, it is clear that addition of structured numbers is not in general commutative.

Multiplication should be defined as a suitable iteration of addition, exponentiation as an iteration of multiplication, *etc.* Such a process is taking its inspiration from what has been done for corresponding integer operations; in that case, a denumerable collection of operations result that lead to the classical Ackermann function of recursive function theory. The talk and the paper [1] both propose to examine what survives of properties of integers in this context.

A whole *hierarchy* of binary operations on trees is introduced recursively as follows:

$$a \cdot^1 b = a + b, \qquad a \cdot^{k+1} b = (a \cdot^k b_0) \cdot^k (a \cdot^k b_1),$$

where b_0, b_1 are the left and right root subtrees of b. We thus have by definition $a \cdot^1 b = a + b$, and we define *multiplication* and *exponentiation* by

$$a \times b = a \cdot^2 b, \qquad a^b = a \cdot^3 b.$$

Note that the multiplication in $a \times b$ can be viewed as the process of grafting copies of a at each leaf of b, that is to say, as the substitution $b[a]$.

THEOREM 1 (WEIGHT THEOREM). *With a and b arbitrary trees, one has*

$$|a + b| = |a| + b|, \quad |a \times b| = |a| \cdot |b|, \quad |a^b| = |a|^{|b|}.$$

For higher-order operations, the weight of the result is no longer independent of the shape of the operands.

THEOREM 2 (DISTRIBUTIVITY). *With a, b, c arbitrary trees, one has*

$$a \cdot^k (b + c) = (a \cdot^k b) \cdot^{k-1} (a \cdot^k c), \quad a \cdot^k (b \times c) = (a \cdot^k b) \cdot^k c.$$

For instance, we have the natural generalizations

$$a \times (b + c) = a \times b + a \times c, \; a \times (b \times c) = (a \times b) \times c,$$

$$a^{(b+c)} = a^b \times a^c, \; a^{b \times c} = \left(a^b\right)^c.$$

These two theorems are representative of the results of [1]. Other properties include right simplifiability of $+, \times$ (the property stops at level 3 of the hierarchy!).

2. Prime trees

It is clear from the interpretation of multiplication as substitution that a tree is composite or non-prime (is non-trivially decomposable under multiplication) if and only if one of its "fringes" consists of identical trees. Each tree then factors uniquely into primes [1]. From there, it is natural to ask whether there is some sort of a prime density theorem for structured numbers. The answer was obtained jointly by the speaker and the author of this summary. We briefly explain it here.

The number T_n of all trees of size n is known and given by (1). Let I_n be the number of those that are primes; clearly, we must adopt $I_1 = 0$ (1 is not a prime!). Let $T(z)$, $I(z)$ be the corresponding generating functions:

$$T(z) = \sum_{n \geq 1} T_n z^n = z + z^2 + 2z^3 + 5z^4 + 14z^5 + 42z^6 + \cdots,$$

$$I(z) = \sum_{n \geq 1} I_n z^n = z^2 + 2z^3 + 4z^4 + 14z^5 + 38z^6 + \cdots.$$

Combinatorial classics [2] teach us that

$$T(z) = \frac{1 - \sqrt{1 - 4z}}{2}.$$

Now decomposing trees according to their prime "trailers" yields a relation defining $I(z)$ implicitly:

(2) $$T(z) = z + \sum_{k=2}^{\infty} T_k I(z^k), \qquad T_n = \delta_{n,1} + \sum_{d|n, d \geq 2} T_{n/d} I_d.$$

We recognize here a product of (formal) Dirichlet series. Setting

$$\tau(s) = \sum_{n=1}^{\infty} \frac{T_n}{n^s}, \qquad \iota(s) = \sum_{n=1}^{\infty} \frac{I_n}{n^s},$$

we have the relation matching (2):

$$\tau(s) = 1 + \tau(s)\iota(s) \quad \text{or} \quad \iota(s) = 1 - \frac{1}{\tau(s)}.$$

Thus expanding $1/\tau(s)$ as $(1+v(s))^{-1}$ yields

$$I_n = T_n - \sum_{\substack{d_1 d_2 = n \\ d_j \geq 2}} T_{d_1} T_{d_2} + \sum_{\substack{d_1 d_2 d_3 = n \\ d_j \geq 2}} T_{d_1} T_{d_2} T_{d_3} - \cdots .$$

In particular $T_n - I_n$ is equal to 0 if n is prime (as it should), is equal to $(T_p)^2$ if $n = p^2$ is the square of a prime, and is otherwise approximated by $2T_p T_{n/p}$ if p is the smallest prime divisor of n. Here are a few initial values.

n	1	2	3	4	5	6	7	8	9	10	11	12
T_n	1	1	2	5	14	42	132	429	1430	4862	16796	58782
I_n	0	1	2	4	14	38	132	420	1426	4834	16796	58688

Note that the asymptotic form of T_n results from Stirling's formula:

$$T_n \sim \frac{4^{n-1}}{\sqrt{\pi n^3}}.$$

Clearly, almost trees are irreducible: the asymptotic density of primes is thus 1 and further characterized by the remarks above.

Bibliography

[1] Blondel (Vincent). – Une famille d'opérateurs sur les arbres binaires. *Comptes-Rendus de l'Académie des Sciences*, vol. 321, 1995, pp. 491–494.

[2] Comtet (Louis). – *Advanced Combinatorics*. – Reidel, Dordrecht, 1974.

[3] Knuth (Donald E.). – *The Art of Computer Programming*. – Addison-Wesley, 1968, vol. 1: Fundamental Algorithms. Second edition, 1973.