Zeroless Arithmetic: Representing Integers ONLY using ONE

Edinah K. GNANG and Doron ZEILBERGER

"The One counts Himself, and no-one else counts Him, and He is every number, He is root, and foundation and square and cube, and He is like the essence that carries all the cases, and every number is in His power, and He is in every number in deed, and He is present, and every number is present because of Him, and He is Ancient, and every [other] number is [re]newed, and He is the reason for every number, pair[even] and that is not pair, He is not a number, and will not multiply and will not divide."

- Abraham Ibn Ezra (1089-1164), *Sefer HaEkhad* ("Book of One")[I]

Abstract: We use *recurrence equations* (alias difference equations) to enumerate the number of formula-representations of positive integers using only addition and multiplication, and using addition, multiplication, and exponentiation, **where all the inputs are ones**. We also describe efficient algorithms for the random generation of such representations, and use Dynamical Programming to find a shortest possible formula representing any given positive integer.

Very Important: This article is accompanied by the Maple package

`http://www.math.rutgers.edu/~zeilberg/tokhniot/ArithFormulas` ,

and the output files that are linked to from the webpage ("front") of this article

`http://www.math.rutgers.edu/~zeilberg/mamarim/mamarimhtml/arif.html` .

Prologue

According to conventional wisdom, the invention ("discovery") of *zero* was one of the greatest moments in the annals of mathematics. We respectfully disagree. The invention of zero was a great disaster, that lead to the beginning of *nihilism*. Here we will show how it is possible to manage very well without 0.

Introduction

Mark Twain once wrote a letter to a friend that started with

"I didn't have time to write a short letter so I wrote a long one ..."

We mathematicians (and computer scientists) deal with *numbers* rather than words, but even the seemingly naive question of *representing* a positive integer as succinctly as possible is far from trivial.

This interesting question was addressed in [GD], where the systematic study of arithmetical formula-representation was initiated, and two natural ways, called there "the first canonical form" (FCF), and the "second canonical form" (SCF) were introduced. The present article is a natural follow-up of [GD], but in order to make it self-contained, we will review the basic notions.

We are all familiar with the "caveman's representation" of a positive integer by marking lines (only using 1's), for example

$$17 = 11111111111111111 \quad ,$$

also called the *unary* representation. More "efficiently" we have the familiar decimal, 'positional' systems that, alas, needs **ten** symbols. The binary representation "only" uses two (one too many!) symbols, 0 and 1, where, for example, seventeen is written as 10001, meaning

$$17 = 1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 \quad .$$

One can use the "sparse notation" by *only* keeping the 1's

$$17 = 2^4 + 1 \quad .$$

and doing the same for the exponents

$$17 = 2^{2^2} + 1 \quad ,$$

and finally replacing 2 by $1+1$ getting an expression that **only** uses 1

$$17 = (1+1)^{(1+1)^{1+1}} + 1 \quad .$$

This lead (in [GD]) to the *First Canonical Form.* Another natural way is to use the *Fundamental Theorem of Arithmetic* and factor the integer into prime powers, and then either write each prime as a sum of 1's and keep factorizing the exponents, or write a prime as $1 + (p-1)$ and factorize $p-1$ and continue recursively. This lead, in [GD], to the *Second Canonical Form.*

Either way, the *bottom line* is an *expression* that only uses 1's, plus ("+"), times ("$*$"), and exponentiation ("$\wedge$").

We will make the convention that 1 can never be an argument of either multiplication or exponentiation, or else there would be infinitely many ways of representing even 1.

Given a positive integer n, how can we express it as a *formula* **only** using the operations $\{+, *, \wedge\}$ and the integer 1? [where we consider our operations as *binary*, i.e. *fan-in* 2]

Of course there is only one way to express 1, namely, 1. There is also only one way to express 2:

$$2 = 1 + 1 \quad .$$

[Strictly speaking we should write $2 = (1) + (1)$, but we will abuse notation and abbreviate (1) to 1].

There are exactly two ways to express 3

$$3 = (1+1)+1 \quad , \quad 3 = 1+(1+1) \quad .$$

So far we only used addition. There are five ways to express 4 only using addition:

$$1+((1+1)+1) \quad , \quad 1+(1+(1+1)) \quad , \quad (1+1)+(1+1) \quad , \quad ((1+1)+1)+1 \quad , \quad (1+(1+1))+1 \quad .$$

[In general there are $C_n = (2n)!/(n!(n+1)!)$ ways of expressing n *only* using addition].

If you are also allowing multiplication, then we have, in addition (no pun intended)

$$4 = (1+1)*(1+1) \quad ,$$

and if you are also allowing exponentiation, we have

$$4 = (1+1) \wedge (1+1) \quad .$$

The above are examples of *formulas* whose inputs are always 1's. The easiest way to define a formula is via 'grammars'. If we only use addition, the additive formulas are given by the grammar

$$F = 1 \quad OR \quad (F)+(F) \quad ,$$

while the formulas that allow both addition and multiplication are defined by

$$F = 1 \quad OR \quad (F)+(F) \quad OR \quad (F)*(F) \quad ,$$

and if you also allow exponentiation, then the grammar is

$$F = 1 \quad OR \quad (F)+(F) \quad OR \quad (F)*(F) \quad OR \quad (F) \wedge (F) \quad .$$

The above format is *infix*. As is well known (especially to users of HP calculators) one can get rid of parentheses, using *postfix* (alias Reverse Polish) notation. The translation from infix to postfix is easy

$$1 \to 1 \quad , \quad a+b \to ab+ \quad , \quad a*b \to ab* \quad , \quad a \wedge b \to ab \wedge \quad .$$

Of course these transformation rules are to be applied recursively. For example, the expression

$$(1+(1+1)) \wedge ((1+1)+1)$$

(representing twenty-seven), is written in postfix notation as

$$111++11+1+\wedge \quad .$$

We have already mentioned that the number of expressions of n that only use addition, let's call it $C_a(n)$, is the famous Catalan sequence $(2n)!/(n!(n+1)!$ (why?). Let $C_{am}(n)$ be the number of such expressions that use both addition and multiplication, and $C_{ame}(n)$ the number of expressions that use the full arsenal of addition, multiplication, and **exponentiation**.

In this short article (accompanied by a very long Maple package, and even longer sample output files) we will answer the following questions.

• How to compute the sequences $C_{am}(n)$ and $C_{ame}(n)$ for as many n as possible ?(it is unlikely that there are closed-form formulas).

• What is the asymptotics of $C_{am}(n)$ and $C_{ame}(n)$ as $n \to \infty$?

• How to draw *uniformly at random*, such an expression ?

• How to find the shortest possible expression for a given integer n. Of course, if you only use addition all $C_a(n)$ expressions have the same length $2n-1$, but of course if one allows multiplication one can get much shorter expressions, and if one also allows exponentiation, then one can get yet shorter ones. [The length of such a minimal expression may be called the *computational complexity* of the integer (w.r.t. the computational models discussed here)]

Enumeration

Only using addition

Let $C_a(n)$ be the number of expressions for the positive integer n *only* using addition. Such an expression may be written as $n = k + (n - k)$ for some $1 \le k < n$, and the number of these is $C_a(k)C_a(n-k)$, so we have the non-linear recurrence

$$C_a(n) = \sum_{k=1}^{n-1} C_a(k)C_a(n-k) \quad , \quad C_a(1) = 1,$$

whose solution is famously $(2n)!/(n!(n+1)!)$, the ubiquitous Catalan sequence [S] `http://oeis.org/A000108`.

Using addition and multiplication

Let $C_{am}(n)$ be the number of formula-trees with the leaves all 1's that represent the integer n, and $C^a_{am}(n)$ be the number of those whose root is $+$, and $C^m_{am}(n)$ be the number of those whose root is $*$. Then we have, of course

$$C_{am}(n) = C^a_{am}(n) + C^m_{am}(n) \quad ,$$

and the non-linear recurrences

$$C^a_{am}(n) = \sum_{i=1}^{n-1} C_{am}(i)C_{am}(n-i) \quad ,$$

$$C^m_{am}(n) = \sum_{i>1, n/i \ integer}^{\lfloor n/2 \rfloor} C_{am}(i) C_{am}(n/i) \quad .$$

[See procedures `Cam(n)` and `CamSeq(N)` in `ArithFormulas`].

Using the *Wilf-methodology* [W][NW] we can use the *remembered* values of $C_{am}(n)$ and $C^a_{am}(n)$, $C^m_{am}(n)$ to generate *uniformly at random* such an expression. First use a *loaded* coin with probabilities $C^a_{am}(n)/C_{am}(n)$, $C^m_{am}(n)/C_{am}(n)$ to decide whether the root-operation is "plus" or "times", and in the former case use an $n-1$-faced loaded die whose faces are labeled $1, ..., n-1$, and the probability of lending on i is $C_{am}(i)C_{am}(n-i)/C^a_{am}(n)$, and continue recursively for $i, n-i$ assuming that it landed on i. Similarly if the loaded coin decided that the root-operation is "$*$" then, create a loaded die whose faces are labeled by the non-trivial divisors of n, and the probability of lending on face i is $C_{am}(i)C_{am}(n/i)/C^m_{am}(n)$ and continue recursively.

[See procedures `RaFamT(n)` and `RaFamP(n)`in `ArithFormulas`].

Using addition, multiplication and exponentiation

Let $C_{ame}(n)$ be the number of formula-trees, whose internal nodes are in $\{+, *, \wedge\}$ and whose leaves are all 1's, that represent the integer n, and $C^a_{ame}(n)$ be the number of those whose root is $+$, $C^m_{ame}(n)$ be the number of those whose root is $*$, $C^e_{ame}(n)$ be the number of those whose root is $\wedge$.

Then we have, of course

$$C_{ame}(n) = C^a_{ame}(n) + C^m_{ame}(n) + C^e_{ame}(n) \quad ,$$

and the non-linear recurrences

$$C^a_{ame}(n) = \sum_{i=1}^{n-1} C_{ame}(i) C_{ame}(n-i) \quad ,$$

$$C^m_{ame}(n) = \sum_{i>1, n/i \ integer}^{\lfloor n/2 \rfloor} C_{ame}(i) C_{ame}(n/i) \quad .$$

$$C^e_{ame}(n) = \sum_{i^j=n, j>1} C_{ame}(i) C_{ame}(j) \quad .$$

[See procedures `Came(n)` and `CameSeq(N)` in `ArithFormulas`].

Using the *Wilf-methodology* [W][NW] we can use the *remembered* values of $C_{ame}(n)$ and $C^a_{ame}(n), C^m_{ame}(n), C^e_{ame}(n)$ to generate *uniformly at random* such an expression, in an analogous way to the addition-multiplication trees above.

[See procedures `RaFameT(n)` and `RaFameP(n)`in `ArithFormulas`].

Finding the Shortest Formula

Using *Dynamical programming* we can find the shortest possible formula (measured in terms of length in postfix notation), in either categories. We look at all the possible root operations and their subtrees and pick the shortest possibility, using the previously obtained expressions for the children.

[See procedures `ShortestTam(n)`, `ShortestTame(n)` for the shortest formulas in infix (tree) notation and procedures `ShortestPam(n)`, `ShortestPame(n)` for the shortest formulas in postfix (Reverse Polish) notation].

Asymptotics

The well-known asymptotics for $C_a(n) = (2n)!/(n!(n+1)!)$ can be easily derived from Stirling's formula, yielding $\frac{1}{\sqrt{\pi}}4^n n^{-3/2}$. It is much harder to derive the asymptotics for $C_{am}(n)$ and $C_{ame}(n)$ rigorously, but using procedure `Zinn` of `ArithFormulas`, we get the following non-rigorous estimates

$$C_{am}(n) \asymp c_1 n^{-3/2}(4.077...)^n \quad ,$$
$$C_{ame}(n) \asymp c_2 n^{-3/2}(4.131...)^n \quad ,$$

for some constants c_1, c_2.

The Book of Minimal Formulas

To get the enumeration (up to $n = 40$), and a list of optimal-length formulas for n from 2 to 8000, generated by procedures `SeferAM(K1,K2)` and `SeferAME(K1,K2)` (with $K1 = 40, K2 = 8000$) for formulas using only addition and multiplication and for formulas also using exponentiation, respectively, see the two webbooks

`http://www.math.rutgers.edu/~zeilberg/tokhniot/oArithFormulas1` ,

`http://www.math.rutgers.edu/~zeilberg/tokhniot/oArithFormulas2` .

These minimal expressions are listed in postfix notation, ready to be entered into a Reverse Polish Calculator (available on-line, e.g. `http://www.alcula.com/calculators/rpn/`, viewed March 1, 2013). They are given in the most memory-efficient way (using procedure `MinMemory`) so as to minimize the number of memory locations (stack-size) needed, i.e. realizing the *Strahler number* (see `Stra` in `ArithFormulas`).

We also have analogous procedures for using *addition and exponentiation* (i.e. **no** multiplication). The output is presented in the following webbook

`http://www.math.rutgers.edu/~zeilberg/tokhniot/oArithFormulas3` .

Conclusion

In addition to the great intrinsic interest of this project-what can be more natural or fundamental than expressing integers?-it is also a *case study* in using **Experimental Mathematics** to enumerate, randomly generate, and optimally generate, combinatorial objects. We believe that the same methodology could be applied to *Boolean formulas* and even *Boolean circuits*, that would shed yet another angle on the central problem of theoretical computer science, the notorious P vs. NP problem. So far most of the work was done by humans, using pencil-and-paper. It is about time that computers will put some effort towards settling the most central problem of their field, or at the very least, give some *empirical* and *experimental* insight about it.

References

[GD] Edinah K. Gnang and Patrick Devlin, *Some Integer Formula-Encodings and related algorithms*, to appear in Adv. Appl. Math. [available from `www.arxiv.org`]

[I] Abraham Ibn Ezra, *"Sefer HaEkhad"* [The "Book of One"], available on-line from `http://www.scribd.com/doc/17323522/Ibn-Ezra-Sefer-HaEchad` [viewed March 2, 2013]

[NW] Albert Nijenhuis and Herbert S. Wilf, *"Combinatorial algorithms for computers and calculators"*, Academic Press, 2nd edition, 1978.

[S] Neil Sloane, *The On-Line Encyclopedia of Integer Sequences*, `oeis.org`.

[W] Herbert S. Wilf, *A unified setting for sequencing, ranking, and selection algorithms for combinatorial objects*, Advances in Mathematics **24** (1977), 281-291.

Edinah K. Gnang, Computer Science Department, Rutgers University (New Brunswick), Piscataway, NJ 08854, USA. `gnang at cs dot rutgers dot edu`

Doron Zeilberger, Mathematics Department, Rutgers University (New Brunswick), Piscataway, NJ 08854, USA. `zeilberg at math dot rutgers dot edu`

March 4, 2013