Right associative exponentiation normal forms and properties

Armando B. Matos

first version: 2005
current version: October 12, 2011

Abstract

Consider the representation of integers by arithmetic expressions involving only the right associative exponentiation symbol, without using parenthesis. The existence of a unique exponential normal form, a result somewhat analogous to the unique factorization of a positive integer, is proved. Prime numbers of multiplicative arithmetic $(\mathbb{N}^+, \times)$ correspond to the so called *base numbers* of exponential arithmetic $(\mathbb{N}^+, \uparrow)$.

Some properties of the Kolmogorov complexity of integers represented in the exponential arithmetic are studied.

Keywords. Exponential representation, exponential normal form, shortest representations.

1 Introduction

Consider the representation of integers by expressions involving only exponentiation; no parenthesis are allowed and "$\uparrow$" associates to the right. The inclusion of parenthesized expressions would correspond to a more complex arithmetic; in particular arbitrary *products* of exponent would be possible, for instance $(a \uparrow b) \uparrow ((c \uparrow d) \uparrow e) = a \uparrow (b(c \uparrow (de)))$.

We write the exponential form of the integer n as $n = a_1 \uparrow a_2 \uparrow \cdots \uparrow a_k$, where the a_i's are "base numbers" (to be defined later). For convenience, the usual notation, $a^b \equiv a \uparrow b$ will be sometimes used. There is an equivalent to the Unique Factorization Theorem, see Theorem 2.

Algebra:	$(\mathbb{N}^+, \times)$	$(\mathbb{N}^+, \uparrow)$
	Prime factorization (unique)	Exponential form (unique)
	Prime number	Base number

However, the similitude between the prime factorization and the exponential form is only partial. The main difference is that exponentiation is not commutative, a fact that has important applications.

We study the properties of this *exponential* representation, and the length of the shortest representation of an integer. When the computation model is universal, the length of the shortest representation of an integer is the Kolmogorov complexity of that integer, see [2].

It is interesting to study the properties of more restricted modes of representation like the arithmetic $(\mathbb{N}^+, \uparrow)$; for a similar study using the arithmetic $(\mathbb{N}^+, \times)$ with the unary integer representation see [3]. As every arithmetic expression corresponds to a total function, the "arithmetic Kolmogorov complexity" is decidable; some of its properties, like the incompressibility theorem, are similar to those of the usual Kolmogorov complexity.

The notation used in this paper is fairly straightforward. In particular, $\mathbb{N}$ and $\mathbb{N}^+$ denote the set of integers and the set of positive integers, respectively. The symbol "$\leftrightarrow$" means "corresponds to". The ordered sequence of the integers a, b,... is denoted by $\langle a, b, \ldots \rangle$. The oprrator "$\cdot$" will denote the concatenation of two ordered sequences.

The contents and organization of this paper is as follows. In Section 2 we give a necessary and sufficient condition for the equality $a^x = b^y$ and study exponential normal forms. In particular, we prove that the *exponential normal form* is unique and show a bijection between $\mathbb{N}^+$ and the set of finite sequences of integers that are not perfect powers is described. Then, in Section 3, the shortest exponential expressions denoting a given integer are considered and an efficient algorithm for finding a shortest exponential expression is described. In Section 4 we study the different exponential forms of a given integer. First, we present a few simple rules that generate all the exponential forms of a given number. Then, the density of perfect powers is studied. Finally, we study the number of exponential expressions denoting a given integer and prove that for $n \geq 2$ the number of exponential expressions with value n does not exceed $(\log n)^2$.

2 Exponential normal form

We start with a simple result that characterizes the possibility of writing an integer as two or more exponentials. We have for instance $4^3 = 8^2$. The following theorem characterizes the integers a and b for which there exist x and y such that $a^x = b^y$.

Theorem 1 *Let a and b be positive integers. There exist positive integers x and y such that $a^x = b^y$ if and only if the primes occurring in the prime factorizations of a and b are the same and all the corresponding exponents are in the same proportion.*

Proof. Suppose that $a^x = b^y$ and let p^e be any factor in the prime factorization of a. The same prime p must be a divisor of b, let $p^{e'}$ be the corresponding factor in the prime factorization of b. Clearly we must have $ex = e'y$ or $e/e' = y/x$.

Conversely suppose that

$$a = p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k}, \;\; b = p_1^{e'_1} p_2^{e'_2} \cdots p_k^{e'_k}$$

where, for $1 \leq i \leq k$, e_i/e'_i is the same rational number, say p/q. It can be easily seen that $a^q = b^p$. As an example let

$$a = 2^3 \times 7^6, \; b = 2^5 \times 7^{10}$$

Here $p = 3$, $q = 5$ so that $a^5 = b^3 = 738569102645403913023102943232$. □

We say that an integer n is a *perfect power* is there exist $a \geq 2$ and $b \geq 2$ such that $n = a^b$. When $n \geq 2$ is not a perfect power we say that n is a *base*[1].

Theorem 2 *Any positive integer n can be written in an unique form*

$$n = a_1 \uparrow a_2 \uparrow \cdots \uparrow a_k$$

where every a_i is a base; this form is called the exponential normal form *of n and is denoted by $E(n)$.*

Instead of a formal proof we give a simple recursive algorithm to compute $E(n)$.

INPUT: Integer $n \geq 2$
OUTPUT: $E(n)$, the exponential normal form of n
`expression enf(n)`:
  `let` $p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k}$ be the prime factorization of n
  `let` $a = \gcd\{e_1, \cdots, e_k\}$.
  `if` $a = 1$ `return` n (n is not decomposable)
  `otherwise`
    `return` the expression $m \uparrow E(a)$
      `where` $m = p_1^{e_1/a} p_2^{e_2/a} \cdots p_k^{e_k/a}$.

One obtains for instance, the following normal form of 512

$$512 = 2 \uparrow 3 \uparrow 2$$

In fact we do not need to decompose n in prime factors and $E(n)$ can be computed in polynomial time.

The set of base numbers 2, 3, 5, 6, 7, 10... will be denoted by $\mathcal{B}$.

2.1 Correspondence between $\mathbb{N}^+$ and $\mathcal{B}$

There is a bijective correspondence between integers $n \geq 2$ and nonempty sequence of base numbers $\langle a_1, \cdots, a_k \rangle$:

$$n = a_1 \uparrow a_2 \uparrow \cdots \uparrow a_k$$

[1] This simple terminology is not standard but is adequate for the purposes of this paper.

The correspondence is obviously a bijection: to every integer corresponds an unique nonempty sequence of base numbers (Theorem 2) and reciprocally, to every nonempty sequence of base $\langle a_1, \cdots, a_k\rangle$ corresponds the integer $a_1 \uparrow a_2 \uparrow \cdots \uparrow a_k$.

The correspondence can be extended to a correspondence between $\mathbb{N}^+$ and $\mathcal{B}$ by associating 1 to the empty sequence $\langle\rangle$. Denote by $s(n)$ the sequence corresponding to n.

As an example, $n = 512$ corresponds to the sequence $\langle 2, 3, 2\rangle$.

2.1.1 Sequence corresponding to the product

It is interesting to notice how two sequences can be "multiplied". Let $s(m) = x$ and $s(n) = y$. First write $x = uz$ and $y = vz$ where z is the longest suffix common to x and y. We have

$$s(mn) = s(s^{-1}(u) \times s^{-1}(v)) \cdot z$$

Notice that this rule works even if the longest common suffix equals x or y (or both).

The justification for this rule is simple. Just notice that, if we denote by c the integer represented by the sequence z and write $m = a^c$, $n = b^c$, we get $mn = (a \times b)^c$ whose representation is $s(a \times b) \cdot z$.

Example 3 *To multiply* $16 \leftrightarrow \langle 2, 2, 2\rangle$ *by* $81 \leftrightarrow \langle 3, 2, 2\rangle$ *we get:* $s(16 \times 256) = s(s^{-1}(2) \times s^{-1}(3)) \cdot \langle 2, 2\rangle = \langle 6\rangle \cdot \langle 2, 2\rangle = \langle 6, 2, 2\rangle \leftrightarrow 6^4 = 1296 = 16 \times 256$.

3 On the shortest representations of an integer

In the arithmetic $(\mathbb{N}^+, \uparrow)$ what is the shortest representation of an integer? As we said above, no parenthesis are allowed and the operator "$\uparrow$" associates to the right. We assume that the integers are represented in some basis.

In general expressing a number as a power results in a shorter expression; for instance, $2 \uparrow 64$ is shorter than 18446744073709551616. If the rule "if possible represent an integer by a power" is applied recursively, *the sorter expression representing an integer is its exponential normal form.* However for small integers, and depending on the number basis used and on the symbol lengths, representing an integer n as a^b may result in a longer expression; for instance, if we use 10 as the basis of numeration and measure the length of an expression by the number of symbols it contains, $|8| = 1$ but $|2 \uparrow 3| = 3$. Also, $3 = |256| < 5 = |2 \uparrow 2 \uparrow 3|$. This, as we said above can only happen with small integers.

Which integers are most compressible? Again, if the rule "shorter expressions are the powers" is applied, the most compressible integers[2] are of the form

$$2 \uparrow 2 \uparrow \cdots \uparrow 2$$

which takes us to Knuth's tower notation [1]. Due to the reasons mentioned above, this result may be not exactly true...

3.1 Efficient algorithm for finding a shortest exponential expression

We now describe an algorithm which finds the set of all exponential expressions having a given value n. This set has cardinality polynomial in $|n|$ and the algorithm runs in polynomial (also in $|n|$) time. It follows that a shortest expression representing a given integer can also be found in polynomial time.

```
      function reps(n):
(1)      Input:  n
(2)      Output:  set of exponential expressions of n
(3)           S = {n}
(3)           for b = 2, 3, ··· , ⌊log n⌋
(4)                if ∃a such that a^b = n then
(5)                     S_b = reps(b)
(6)                     S = S ∪ {a ↑ b : b ∈ S_b}
(7)           return S
```

Example 4 *In this example the length of an expression by the number of symbols it contains. Let $n = 256$. The test in line* `(4)` *succeeds for the following values of b*

- *From line* `(1)`*: $S = \{256\}$.*
- *$b = 2$, $a = 16$. From the (recursive) call in line* `(5)` *we get the set of representations of 2: $S_b = \{2\}$; S becomes $\{256, 16 \uparrow 2\}$.*
- *$b = 4$, $a = 4$. From the call in line* `(5)` *we get the set of representations of 4: $S_b = \{4,\ 2 \uparrow 2\}$; S becomes $\{256, 16 \uparrow 2,\ 4 \uparrow 4, 4 \uparrow 2 \uparrow 2\}$.*
- *$b = 8$, $a = 2$. From the call in line* `(5)` *we get the set of representations of 8: $S_b = \{8,\ 2 \uparrow 3\}$; S becomes $\{256, 16 \uparrow 2,\ 4 \uparrow 4, 4 \uparrow 2 \uparrow 2, 2 \uparrow 8, 2 \uparrow 2 \uparrow 3\}$. This set is returned by the function* `reps`*.*

Comment on the correctness. The algorithm above never generates repeated expressions; this is easily seen by induction on $|n|$; it is enough to notice that the values of a obtained in line `(4)` are all different.

[2] That is, integers n such that the ratio (length of a shortest expression with value n)/$|n|$ is as small as possible.

Comment on the efficiency. From the following facts it can easily be shown that the algorithm runs in polynomial time (in terms of $|n|$)

- Each call to `reps` generates at least a new element of S.
- The test in line in line `(4)` can be implemented in polynomial time.
- The number of exponential representations of an integer n is polynomial in $|n|$ (notice that $|n|$ is $O(\log n)$).

4 Exponential forms of an integer

4.1 Rules

In this Section we describe a few simple rules that generate all the exponential forms of a given number.

In the following rules the transformed sub-expressions are on the rightmost part of the global expression; this is what we mean by writing "$[\cdots X]$" where X is some expression. The symbol n denotes an integer. The symbol a also denotes an integer (having some exponent in the global expression). By a^n we mean the corresponding integer, not the expression.

$$
\begin{array}{lrcll}
(1) & [\cdots a \uparrow n] & \longrightarrow & [\cdots a^n] & \\
(2) & [\cdots n] & \longrightarrow & [\cdots a \uparrow m] & (n = a^m,\ m \geq 2 \text{ not a perfect power}) \\
(3) & [\cdots a \uparrow n] & \longrightarrow & [\cdots a^k \uparrow (n/k)] & (k|n, 2 \leq k < n)
\end{array}
$$

Rules (1) and (2) are the inverse of each other. Rule (2) only applies iff n is a perfect power.

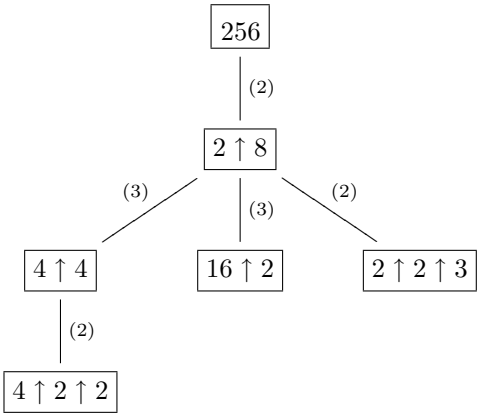
We can generate all exponential expressions by starting with the integer n as root and

1. Apply rule (2), obtaining $a \uparrow m$.
2. For each divisor d of m with $2 \leq d \leq m$ apply rule (3).

Then reapply the process to each rightmost exponent not yet considered.

As an example, the diagram below contains all exponential expressions with value 256. Notice that rule (1) is never used because it would result in the

immediate ancestor ("mother"or "father") of the node in consideration.



4.2 On the density of perfect powers

The number of perfect powers not exceeding n is approximately $\sqrt{n}$. To see this, notice that the number of perfect powers of exponent e not exceeding n is $\lfloor n^{1/e} \rfloor$. Furthermore[3] we only have to consider prime exponents for, if e is not prime, perfect powers with exponent e are also perfect powers with a smaller (and prime) exponent. Thus, the number $P(n)$ of perfect powers not exceeding n satisfies

$$P(n) \leq \Sigma_p \lfloor n^{1/p} \rfloor \leq \Sigma_p \, n^{1/p}$$

where in the summations over p it is assumed that p is prime. We get

$$\frac{P(n) - \sqrt{n}}{\sqrt{n}} \leq \Sigma_{p \geq 3} \;\; n^{1/p - 1/2} \leq \Sigma_{p \geq 3} \;\; n^{-1/6}$$

In fact this sum is finite; the greater possible exponent p satisfies $3^p \leq n$ or $p \leq \log_3 n$. Thus

$$\frac{P(n) - \sqrt{n}}{\sqrt{n}} \leq \frac{\log_3 n}{n^{1/6}} < \frac{\ln n}{n^{1/6}}$$

so that

$$\lim_{n \to \infty} \frac{P(n) - \sqrt{n}}{\sqrt{n}} = 0 \tag{1}$$

In conclusion, the number of perfect powers less than or equal to n is approximately $\sqrt{n}$, the relative error[4] converging to 0 as $n \to \infty$.

[3] This remark is not essential for establishing (1).

[4] Usage of the Prime Number Theorem and a more careful analysis would result in a smaller error term; however, our main goal is to prove (1).

4.3 Number of exponential expressions

We study the number of exponential expressions with value n; for instance there are 6 exponential forms with value 256:

$$2 \uparrow 2 \uparrow 3,\ 2 \uparrow 8,\ 4 \uparrow 2 \uparrow 2,\ 4 \uparrow 4,\ 16 \uparrow 2,\ 256$$

We conjecture that $c(n) \leq \log n$ *holds for every integer* n. This has been verified for every perfect power n not exceeding 1100 and for some other larger integers.

The following result shows that the number of exponential expressions with value n is $O((\log n)^2)$. This is an important result because it has as a consequence that various algorithms that consider every exponential expression with a given value n are polynomial.

Theorem 5 *For every integer* $n \geq 2$ *the number of exponential expressions with value* n *does not exceed* $(\log n)^2$.

Proof. Considering the case where n has more than one exponential form, write n as $n = b^e$ where b is not a perfect power. We have $e \leq \log n$ and for each divisor d of e we can write the expression $n = (b^d) \uparrow (e/d)$, the case $d = e$ corresponding to n itself. Each of these expressions may through its exponent e/d generate other expressions with value n. For every[5] $n \geq 8$ the number of divisors of n including n does not exceed $n/2$. Moreover it can easily be checked that for every $n \in [8, 2^8)$ the number of exponential expressions with value n does not exceed $(\log n)^2$. Thus the following recurrence defines a upper bound $f(n) \in \mathbb{R}$ for the number of exponential expressions with value $n \geq 8$.

$$\begin{cases} f(n) = (\log n)^2 & \text{for } 8 \leq n < 2^8 \\ f(n) = \frac{\log n}{2} f(\log n) & \text{for } n \geq 2^8 \end{cases}$$

Notice that the recurrence defines the value of $f(n)$ for every $n \geq 8$; the number of inductive steps needed to define $f(n)$ is k, the index of the interval containing n:

$$I_0 = [8, 2^8),\ I_1 = [2^8, 2^{2^8}), \cdots$$

Notice also that $f(n)$ is a non-decreasing function of n, a fact easily proved by induction on the index of the interval containing n.

The proof of this Theorem is also by induction on the index k of the interval containing n. The values $n = 2, \cdots, 7$ are easily verified separately.

For $k = 0$ we have $n \in [8, 2^8)$ and the statement of the theorem is easy to check.

For the inductive step we use the inductive assumption

$$f(n) = \frac{\log n}{2} f(\log n) \leq \frac{\log n}{2} (\log \log n)^2$$

[5] This also holds for every $n \geq 4$; the number of divisors equals $n/2$ for $n/12$. We use here the lower bound 8 just for convenience.

We have to prove that $f(n) \leq (\log n)^2$. But this is true if

$$\begin{array}{rlcll} & \frac{\log n}{2}(\log\log n)^2 & \leq & (\log n)^2 & \\ \text{true if} & (\log\log n)^2 & \leq & 2\log n & (\text{for } n \geq 4) \\ \text{true if} & (\log\log n)^2 & \leq & (\log n)^2 & \\ \text{true if} & \log\log n & \leq & \log n & \end{array}$$

But $\log\log n \leq \log n$ for every $n \geq 2$. This completes the proof. □

References

[1] Donald E. Knuth. Mathematics and Computer Science: Coping with Finiteness. Science, **194**, 4271, pp 1235–1242, 1976.

[2] Ming Li and Paul M. B. Vitányi. An Introduction to Kolmogorov Complexity and its Application. Third edition, Texts and Monographs in Computer Science. Springer-Verlag, 2008.

[3] Armando B. Matos. Kolmogorov complexity in multiplicative arithmetic. Technical Report, Departamento de Ciência de Computadores, Universidade do Porto, 2005.