Lossless Geometric Compression

Gleb Rusiaew

14/04/19

We can represent any number a, $a \in N_0$ as $a - [\sqrt[n]{a}]^n$, because:

$$[\sqrt[n]{a}]^n + (a - [\sqrt[n]{a}]^n) = [\sqrt[n]{a}]^n - [\sqrt[n]{a}]^n + a = 0 + a = a$$

Denote by $[x]$ the largest integer not exceeding x Next, define new variable $b_1 \equiv a$, then we redefine:

$$b_i \equiv b_{i-1} - [\sqrt[n_{i-1}]{b_{i-1}}]^{n_{i-1}}$$

and redefine again. As a result, we obtain a finite set that will continue until

$$((b_i < 2) \& (i < 100)) \parallel (\sqrt[n_i]{b_i} \in Z)$$

The second approval follows from a computing power and can be changed depending on the power of your computer. Here is the final view of the set:

$$[\sqrt[n_1]{b_1}]^{n_1} + (b_i - [\sqrt[n_i]{b_i}]^{n_i}), b_i \equiv b_{i-1} - [\sqrt[n_i - 1]{b_i - 1}]^{n_i - 1}$$

To compress the integer, you must write number as set with elements $\sqrt[n_1]{b_1}, (b_i - [\sqrt[n_i]{b_i}]^{n_i}), b_i, n_i$

Any compressed number can be extracted. Prove that. To obtain the initial number, you must solve the expression $[m_1]^{n_1} + (b_i - [\sqrt[n_i]{b_i}]^{n_i})$. Denote by m set $\{\sqrt[n_1]{b_1}, (b_i - [\sqrt[n_i]{b_i}]^{n_i}), n_1\}$. We present result as a binary file and voilà.

Here you can see sources on Python2.7: `https://gist.github.com/rusyaew/a793a954356cbd75070118ada8db8d82`