Nagual and Tonal Maths:
Numbers, Sets, and Processes

Lukas A. Saul
Oxford, England
May, 2015

Abstract:

Some definitions and elementary theorems are given here describing tonal and nagual numbers, sets, and processes.

Mea Culpa

The bulk of this work consists of a rebranding of computable and uncomputable numbers as tonal and nagual numbers. The intent is that armed with a different vocabulary, we will be motivated to return to the fields of numbers before us and try to improve our understanding of their categories. So rather than "there's nothing new under the sun", the reader is encouraged to think "let's take another look anyway".

Definition 1

A *nagual* number is one which has no finite representation.

Such a number is not computable, and displays Martin-Löf randomness. That is, its digits form an algorithmically random sequence. More formally:

Definition 2

Consider $n \in \Re \& -1 < n < 1$

if $\forall p \in \mathbb{N} \exists i \in \mathbb{N} s.t. Comp(p,i) \neq A_i$

where $n = \sum_{i=1}^{\infty} A_i * 10^{-i}$ and we say A_i is the i^{th} digit of *n*,

and $Comp(p,i)$ is the i^{th} output of a universal Turing machine running instruction set *p*,

then *n* is *nagual*.

By suitably extending the range of the indices *i* in Definition 2 we can easily extend the definition for *n*<-1 or *n*>1 (see Definition 5).

Definition 3

A tonal number is one which has a finite representation.

Definition 4

if $\exists p \in \mathbb{N} s.t. \forall i \in \mathbb{N} Comp(p,i) = A_i$.

where $n = \sum_{i=1}^{\infty} A_i * 10^{-i}$ and we say A_i is the i^{th} digit of *n*,

and $Comp(p,i)$ is the i^{th} output of a universal Turing machine running instruction set *p*,

then *n* is *tonal*.

Definition 5

Consider $z \in \mathbb{C}$

if $\forall p \in \mathbb{N} \exists j \text{ or } k \in \mathbb{N} s.t. Comp(p,j) \neq A_j \text{ or } Comp(p,k) \neq B_k$

where $z = \sum_{j=-\infty}^{\infty} A_j * 10^j + i * \sum_{k=-\infty}^{\infty} B_k * 10^k$

and we say A_j, B_j are the j^{th} complex digits of *z*,

and $Comp(p,i)$ is the i^{th} output of a universal Turing machine running instruction set *p*,

then z is *nagual*.

That is to say, if a complex number *z* has either its real or imaginary component nagual, we say it also is nagual. The double sided infinite sum is used here for simplicity, these could also be written from 0 using an appropriate change of variables.

Do nagual numbers exist in $\Re$? The first thing to note is that if they don't exist, then then the cardinality of $\Re$ is no longer $\aleph_1$; for the list of programs *p* is of cardinality $\aleph_0$.

Theorem 1

If $\|\Re\| > \aleph_0$ then $\exists n \in \Re$ which are nagual.

The statement holds for not only the reals but any set $\Re$ for which the cardinality is larger than that of the natural numbers, $\aleph_0$. An example might be the power set of the natural numbers.

Theorem 2

The set of all tonal numbers can be put in a one to one correspondence with a subset of the integers.

It is worth asking for a rigorous definition of the object $Comp(p, i)$. Any suitable language can be chosen for interpreting a program *p*. There exist perhaps more suitably compact languages with only a few symbols needed, known as Turing tarpits. However even a high level language can do the trick of making a one to one correspondence between the set of all programs that produce streams of digits A_i, and a subset of the integers. The canonical procedure is to map an integer *p* to an output $Comp(p, i)$ is as follows:

```
//
//Comp(p,i) script
decompress(p)->p.src
compile(p.src)->p.bin
execute(p.bin, i)
```

For most integers *p*, this will fail to decompress, fail to compile, or fail to execute and return a valid integral result. However one can point to other integers *p* which produce from this $Comp(p, i)$ for example the i^{th} digit of pi, rather tidily using the Bailey–Borwein–Plouffe formula.

We might note that searching for integers which are compressed source code representing real numbers is unfeasible. For example, there is no way to know a priori if a given computation $Comp(p, i)$ which reaches execution will even halt.

For these reasons it could be impossible to prove some numbers to be nagual. However, it is easy to disprove that a number is nagual. One must only demonstrate that the number is tonal, that is to show a finite computer program the represents the number exactly.

Definition 6

A *nagual set* is a set of elements which has no finite representation.

The elements of a nagual set themselves can be of any class. For example, one might consider nagual sets of integers. If we claim that the power set of the integers is uncountably infinite, then we must admit that this power set contains nagual sets. If uncountable, it contains sets other than those which could be represented by finite computer programs.

Just because a set contains nagual numbers does not mean it is uncountable. There can be finite and countably infinite sets which contain nagual numbers. If we take two nagual numbers n_1 and n_2 we can form a finite set like $\{n_1,n_2\}$, or a countably infinite set $\{n_1, 2n_2, 3n_1, 4n_2, ...\}$.

Definition 7

A process is considered nagual if its instruction set is not possible to represent in finite space.

A running nagual process is a running process whose instruction set is necessarily infinite, which is to say it has no beginning and if allowed to continue has no end. More precisely: the nagual process is one for which there is no finite program or instruction set that can be represented in finite time which duplicates its behavior. An oracle is also a nagual process.

One utility of the concept of a nagual process may lie not in actual implementation but in approximation, in a similar manner to the way one approximates a wire as an infinitely long line of current density for the purposes of calculating electric fields.

Examples

1) Speed of light (c) -

   Tonal. Don't be fooled by the dimension (meters/second). This number is a definition, it is specified exactly and hence is tonal.

2) Ratio of frequencies of a Pythagorean Perfect Fifth -

   Tonal. This is the rational number 3/2.

3) Ratio of frequencies of a Even Tempered Fifth -

   Tonal. This is the irrational but tonal number $2^{7/12}$.

4) Ratio of frequencies of a Fifth played by any real instrument -

   Nagual (but without proof). It's unclear your instrument could be perfectly measured or simulated in finite time.

5) γ, the Euler-Mascheroni constant -

   Tonal. While no proof exists that this number is transcendental, it is a perfectly well defined tonal number.

6) α, the fine structure constant

   Nagual (but without proof). It is still possible that some physical constants such as Plank's constant will wind up being tonal but at the moment these are measured values. Same with G the gravitational constant.

There is much more to be said about the different classes of machine $Comp(p,i)$ which we have used in these definitions. One way to describe the limitations of $Comp(p,i)$ is that it's execution must be a nagual process. For example, one might try to construct a new diagonal number $\mathfrak{N}$ not in the set of all tonal reals by creating a program:

```
// return a digit of N
input i
return Comp(i,i)/2
```

This new number $\mathfrak{N}$ differs from the output of every program at the i^{th} digit by a factor of 2. Therefore, it is nagual. However, here is the program right here, it must be tonal. What happened?

What happened is that we have assumed that we can simulate the output $Comp(p, i)$ in a new program. In fact we don't know a priori when this machine will halt or cycle forever. For the purposes of constructing a program to represent a number, this does not represent a finite instruction set.

This leaves only one option, which is that $\mathfrak{N}$ is nagual. In fact the way we have constructed $\mathfrak{N}$ is very similar to the way one might define a program that solves the halting problem. Consider another number $\mathfrak{M}$ whose digits are defined as follows:

```
// return a digit of M
input i
if (Comp(i,1)!=null) return 1
else return 0
```

This program produces a number $\mathfrak{M}$ whose i^{th} digit is 1 if *i* itself is a program that halts and 0 otherwise.

Other examples of nagual numbers are Chaitin's constants. The busy beaver series is a nagual series.

Discussion

> "What, then, is the *nagual?* The *nagual* is the part of us which we do not deal with at all. The *nagual* is the part of us for which there is *no description* - no words, no names, no feelings, no knowledge. The *nagual* is not experience or intuition or consciousness. Those terms and everything else you may care to say are only items on the island of the *tonal*. The *nagual,* on the other hand, is only effect. The *tonal* begins at birth and ends at death, but the *nagual* never ends. The *nagual* has no limit. I've said that the *nagual* is where power hovers; that was only a way of alluding to it. By reasons of its effect, perhaps the *nagual* can be best understood in terms of power."

– Carlos Casteneda,

"Nagual" and "tonal" are derived from Nahuatl words, for which there are some different meanings attributed in various literature. It is hoped that their incorporation here will not only help to elucidate the concepts of computable and uncomputable numbers, but also that the study of these numbers can help to elucidate the concepts of the nagual and the tonal. One coherent thread amongst their meanings is that the nagual is associated with something that is somehow beyond the human experience. We might say that one cannot fit it all in the head. The ideas of uncountability and uncomputability also fit nicely into this framework.