Progress in Biophysics and Molecular Biology 119 (2015) 368–381

Contents lists available at ScienceDirect

Progress in Biophysics and Molecular Biology

journal homepage: www.elsevier.com/locate/pbiomolbio

The universal numbers. From Biology to Physics

Bruno Marchal

IRIDIA, Université Libre de Bruxelles, Belgium

ARTICLE INFO

Article history:
Available online 2 July 2015

RÉSUMÉ

I will explain how the mathematicians have discovered the universal numbers, or abstract computer, and I will explain some abstract biology, mainly self-reproduction and embryogenesis. Then I will explain how and why, and in which sense, some of those numbers can dream and why their dreams can glue together and *must*, when we assume computationalism in cognitive science, generate a phenomenological physics, as part of a larger phenomenological theology (in the sense of the greek theologians). The title should have been *"From Biology to Physics, through the Phenomenological Theology of the Universal Numbers"*, if that was not too long for a title. The theology will consist mainly, like in some (neo)platonist greek-indian-chinese tradition, in the *truth* about numbers' relative relations, with each others, and with themselves. The main difference between Aristotle and Plato is that Aristotle (especially in its common and modern christian interpretation) makes reality WYSIWYG (What you see is what you get: reality is what we observe, measure, i.e. the natural material physical science) where for Plato and the (rational) mystics, what we see might be only the shadow or the border of something else, which might be non physical (mathematical, arithmetical, theological, …).

Since Gödel, we know that Truth, even just the Arithmetical Truth, is vastly bigger than what the machine can rationally justify. Yet, with Church's thesis, and the mechanizability of the diagonalizations involved, machines can apprehend this and can justify their limitations, and get some sense of what might be true beyond what they can prove or justify rationally.

Indeed, the incompleteness phenomenon introduces a gap between what is provable by some machine and what is true about that machine, and, as Gödel saw already in 1931, the existence of that gap is accessible to the machine itself, once it is has enough provability abilities. Incompleteness separates truth and provable, and machines can justify this in some way.

More importantly incompleteness entails the distinction between many intensional variants of provability. For example, the absence of reflexion ($beweisbar(\ulcorner A \urcorner) \rightarrow A$ with *beweisbar* being Gödel's provability predicate) makes it impossible for the machine's provability to obey the axioms usually taken for a theory of knowledge.

The most important consequence of this in the machine's possible phenomenology is that it provides sense, indeed arithmetical sense, to intensional variants of provability, like the logics of provability-and-truth, which at the propositional level can be mirrored by the logic of provable-and-true statements ($beweisbar(\ulcorner A \urcorner) \wedge A$). It is incompleteness which makes this logic different from the logic of provability. Other variants, like provable-and-consistent, or provable-and-consistent-and-true, appears in the same way, and inherits the incompleteness splitting, unlike $beweisbar(\ulcorner A \urcorner) \wedge A$. I will recall thought experience which motivates the use of those intensional variants to associate a knower and an observer in some canonical way to the machines or the numbers.

We will in this way get an abstract and phenomenological theology of a machine M through the true logics of their true self-referential abilities (even if not provable, or knowable, by the machine itself), in those different intensional senses.

Cognitive science and theoretical physics motivate the study of those logics with the arithmetical interpretation of the atomic sentences restricted to the "verifiable" (Σ_1) sentences, which is the way to study the theology of the *computationalist* machine. This provides a logic of the observable, as expected by the Universal Dovetailer Argument, which will be recalled briefly, and which can lead to a comparison of the machine's logic of physics with the empirical logic of the physicists (like quantum logic). This leads also to a series of open problems.

http://dx.doi.org/10.1016/j.pbiomolbio.2015.06.013

1. The discovery of the universal numbers

It all begun with Cantor set theory. Galilee and Gauss were already aware that the function which sends each non negative integers n on $2n$ was a bijection, that is a one–one correspondence, although they did not use this terminology. They saw that infinite sets, like N, can be put in such a bijective correspondence with a subset of themselves, and concluded that we should better avoid to make such infinite set into actual infinite mathematical objects. This will remain so until Cantor, and it is not without courage that Cantor will reconsider this question and accept such infinite sets as legal citizen of the mathematical inquiry. Using a naive notion of set, Cantor will show the existence of bijection between the sets N (natural numbers, non negative integers), and Z (integers), and Q (rational numbers), and will discover that not all infinite set can be put in that bijective $1-1$ correspondence. Indeed Cantor is famous for proving that R, the set of real numbers, or (equivalently) the set of infinite binary sequences, or the set of functions from N to {0,1} or from N to N, are not enumerable, where enumerable means that there is a bijection, or a surjection (if we allow repetition) from N to that set. This proof plays an important role in the mathematical discovery of computers, or universal numbers, as I will briefly illustrate.

Cantor theorem: There is no bijection between N and the set of functions from N to N.

Proof. Proof Let us suppose that there is a bijection from N to the set of functions from N to N. Let us denote by f_i the function which is the image of i by that bijection. Then we can introduce the *diagonal* function g which sends n to $f_n(n)+1$, that is, $g(n)=f_n(n)+1$. That is what I will call the first diagonalization act. g is obviously a function from N to N. Could g belong to the list $f_0,f_1,f_2,\ldots$? Well, if it could there would be some f_k, such that $g=f_k$. In that case, by definition of the equality of function, we have that $f_k(x) = g(x)$ for all x, and in particular $f_k(k) = g(k)$. Applying f_k on itself, or on some description of itself (which here will always be represented by some index number), is what I will call the second diagonalization act.

Now, $g(k)$ is equal to $f_k(k)+1$, by definition of g. So we have, by Leibniz identity rule, that $f_k(k)=f_k(k)+1$. As each f_k is supposed to be a well-defined function, $f_k(k)$ is a number, and by subtracting it on both side of the last equation, we get $0=1$. Thus we can conclude that such a bijection cannot exist.

Now, Cantor will generalize this procedure and will obtain many more similar results in set theory. Yet, difficulties will appear, like Galilee and Gauss did warn us. In fact defining a set by some properties leads to paradoxes, the most known being Russell's paradox. Let us write XY for the statement that X belongs to Y, then let us define the set E of all sets which does not belong to themselves: XE iff ¬ XX, then we get EE iff ¬ EE.

In front of such paradoxes, there will be mainly three reactions by the mathematicians. One will be the impetus to formalize the notions, in a way which avoids the paradoxes. This will lead to a variety of set theories (Zermelo–Fraenkel, Quine New Foundations, Von Neumann Bernays Gödel, etc). A second reaction will be more radical, and will throw away the use of the excluded middle principle (Brouwer's Intuitionism), and the last one, which can be related in many ways to the preceding one, will be an attempt to work on sets which are not too much large, and in particular to try, when possible, to restrict oneself to computable or constructive notions. This will lead many mathematicians to define what is a computable function, which is one step toward the discovery of the universal machine, or universal number. So what could be a computable function?

The intuitive idea is that a function from N to N is computable when we can describe in a finite time, with a non ambiguous language L, how to compute it, in a finite time, on each (finite) input. Such description are called algorithm or procedure.

But is there a universal language capable to describe all computable functions from N to N? When Alonzo Church claimed that his "Lambda Calculus" provides such a universal language, his student Kleene was at first quite skeptical, and he tried to refute that claim by using Cantor's diagonal procedure (which enabled already Gödel to show that there is no universal *provability* system). Indeed, thought Kleene, if a universal language L (universal with respect of defining the notion of computable function) exists, then we know that such a set of computable functions has to be enumerable. The reason is that the finite descriptions of the procedures can be listed. Indeed, they are non ambiguously described, and thus the description have a simple checkable grammar, and so we can order them by length—and for those having the same length, we can sub-order them by alphabetical order, assuming some primitive order on the (finite) alphabet of the language. So if a universal language L exists, we would have an enumeration $\varphi_0,\varphi_1,\varphi_2,\ldots$ of all computable functions from N to N.

All right, but then we can define (again) a function g such that $g(n)=\varphi_n(n)+1$. All φ_n are computable, and "adding one" is without doubt computable. So g should be computable, and should admit a procedure and thus some description in the supposedly universal language L. This means that there is a k such that $g=\varphi_k$, and thus, in particular $g(k) = \varphi_k(k)$, and again $\varphi_k(k) = \varphi_k(k)+1$. All φ_i are computable, so $\varphi_k(k)$ is a number that we can again subtract from both sides of the last equation, and get $0=1$.

Now, it looks we are in trouble. We have certainly not prove that the set of computable functions is not enumerable, as the set of all strings in the alphabet, and the subset of the grammatically correct strings (describing procedures) are clearly enumerable by the argument given above. So it looks like Kleene has simply refuted Church's claim that his language, or any language, can describe *all* computable functions. But looking more closely, Kleene will understand that he has not done that. Kleene's proof just proves that there is no universal language L computing all *and ONLY all* computable functions from N to N. In particular, if L is built in such a way that all procedures compute functions from N to N, then indeed L is not universal. But, this shows that IF a universal language exists, then it *must* also computes *other* things. I guess Kleene already knew that the lambda-calculus expression$(\lambda x.xx)(\lambda x.xx)$ gives a non terminating procedure, and those "other things" will be of that type. This saves the consistency of Church's claim: the apparent paradox in $\varphi_k(k) = \varphi_k(k)+1$ does not obtained, because the computation of $\varphi_k(k)$ will just not terminate. In the computer's jargon, $\varphi_k(k)$ crashes the machine. Kleene said that overnight, after having been skeptical, he will become an ardent defender of Church's thesis. Indeed, he gave the most conceptual and profound argument in favor of the Church's thesis: the closure of the set of partial computable functions for Cantor's transcendental diagonalization procedure, where a partial computable function is now a function from a subset (perhaps equal to N, or empty) to N. We will say that a function from N to N is *total* if it is defined on all numbers, and we will use the term partial function if its domain is a subset of N. Partial functions generalize the notion of function, usually considered total on their domain.

Cantor showed that the set of functions from N to N is not enumerable, Kleene did show that the subset of *total* computable functions, although enumerable, is not computably or recursively enumerable. If this seems a bit weird, keep in mind that although a subset cannot be bigger than the set in which it is a subset, it can be more complex, like the painting of the Mona Lisa is more complex than the paper area on which it is painted, or like the Mandelbrot set, a subset of the complex plane C, looks much more complex (apology for the pun) than C.