The New Promise of Analog Computation

José Félix Costa [1,2*], Bruno Loff [2], and Jerzy Mycka [3]

[1] Department of Mathematics, Instituto Superior Técnico
Universidade Técnica de Lisboa
Lisboa, Portugal
fgc@math.ist.utl.pt
[2] Centro de Matemática e Aplicações Fundamentais do Complexo Interdisciplinar
Universidade de Lisboa
Lisbon, Portugal
bruno.loff@gmail.com
[3] Institute of Mathematics,
University of Maria Curie-Skłodowska
Lublin, Poland
Jerzy.Mycka@umcs.lublin.pl

Abstract. We show that, using our more or less established framework of *inductive definition of real-valued functions* (work started by Cristopher Moore in [9]) together with ideas and concepts of standard computability we can prove theorems of Analysis. Then we will consider our ideas as a bridging tool between the standard *Theory of Computability (and Complexity)* on one side and *Mathematical Analysis* on the other, making real recursive functions a possible branch of *Descriptive Set Theory*. What follows is an Extended Abstract directed to a large audience of CiE 2007, Special Session on Logic and New Paradigms of Computability. (Proofs of statements can be found in a detailed long paper at the address http://fgc.math.ist.utl.pt/papers/hierarchy.pdf.)

1 Statement of the conjecture and its solution

Consider a class of real-valued functions closed under the operations of composition, of finding the solution to a first order differential equation and the taking of an infinite limit. Thinking briefly about the last two operations, one may observe that they seem to be related. For instance,

$$\exp(x) = \lim_{y\to\infty} (1 + \frac{x}{y})^y,$$

and also

$$\exp(0) = 1,\ \partial_y \exp(y) = \exp(y).$$

The number π can be expressed by a differential equation that gives arctan, since $\pi = 4\arctan(1)$, and we also know, e.g., that

$$\pi = \lim_{y\to\infty} \frac{2^{4y+1} y!^4}{(2y+1)(2y)!^2}.$$

* Corresponding author

Many other examples may lead us to wonder if this property is universal, i.e., if we can replace the taking of an infinite limit of a function f by the solution of a first order differential equation involving functions no more complex than f. We may also wonder if there is a limit of definability in Analysis, e.g., to know if via limits we can always define new functions or else if all functions can be defined using an upper bound in the number of limit taking.

We will use the toolbox of computability theory to show that while we can always express the solution of a first order differential equation through infinite limits, we cannot always do the opposite.

2 The model of recursive real-valued functions

In a sequence of papers, starting with Cristopher Moore's seminal paper [9], we have established a robust framework to think about a theory of definability of real-valued functions. This theory covers a large spectrum of functions from classes of recursive functions extended to the real numbers to the characteristic functions of predicates of the Analytic Hierarchy.

An in-depth overview of the achievements of this theory can be studied in our reference papers [4, 11, 12, 14, 8, 15] together with a most recent one by Bruno Loff (see [7]) submitted to this Conference, and [2, 3] for other, no less relevant contributions (and the new trend represented by several recent papers by Olivier Bournez, Manuel Campagnolo, Daniel Graça and Emmanuel Hainry).

In the original paper by Cristopher Moore, the key idea we acknowledge nowadays (among all motivations that such a paper provided) is the replacement of the standard recurrence scheme for recursive functions by the so-called *differential recursion scheme*. In its simplest form (removing the vector formulation) this scheme reads as follows: the $(n+1)$-ary function h is defined from a n-ary function f and a $(n+2)$-ary function g

$$h(x_1, \ldots, x_n, 0) = f(x_1, \ldots, x_n),$$
$$\partial_y h(x_1, \ldots, x_n, y) = g(x_1, \ldots, x_n, y, h(x_1, \ldots, x_n, y)),$$

as the solution of this first-order differential equation, if some conditions hold, where x_1, ..., x_n are the *parameters*, y is the *recurrence variable*, and the last variable of g stands for the *transport variable*.

In 2004, we introduced in [12] the *limit scheme* as a replacement for classical *minimalization*: the n-ary function h is defined from a $(n+1)$-ary function f via infinite limit taking

$$h(x_1, \ldots, x_n) = \lim_{y \to \infty} f(x_1, \ldots, x_n, y).$$

The definition of *Real Recursive Functions* runs now semi-formally as follows:

Definition 2.1. *The class of* Real Recursive Functions*, $REC(\mathbb{R})$ for short, is the smallest class of real-valued functions which contains some constants (-1, 0 and 1 suffice) and the standard projections and which is closed under composition, differential recursion and the taking of infinite limits.*

By now, many readers know the nice starting examples which enchant our eyes due to their simplicity, such as

$$h(x,0) = x,\ \partial_y h(x,y) = 1,$$

having the function of addition $\lambda xy.\ x + y$ as solution, or

$$h(x,0) = 0,\ \partial_y h(x,y) = x,$$

which gives $\lambda xy.\ x \times y$, and

$$h(0) = 1,\ \partial_y h(y) = h(y),$$

resulting in the *exponential*.

Another class of interesting examples uses infinite limits:

$$\delta(x) = \lim_{y \to \infty} (\frac{1}{x^2 + 1})^y$$

which is Kronecker's δ function over the reals, or

$$sgn(x) = \lim_{y \to \infty} \frac{arctan(xy)}{\frac{\pi}{2}}$$

which is the *signal* function, or

$$\Theta(x) = \frac{\delta(x) + sgn(x) + 1}{2},$$

the Heaviside function defined by composition.

A real recursive number in our framework is the value of real recursive function on a basic constant like 0. Notice that the class of real recursive functions is countable infinite, thus the set of real recursive numbers is also countable. It turns out that a number $y = h(x)$, where x is a previously defined real recursive number and h some real recursive function, is also real recursive. E.g., Neper's e is given by $\exp(1)$ and π is given by $4 \arctan(1)$. Numbers can be thought of as entire computable structures, indivisible entities [9], or computable by digits (as in the classical way), using continued fractions.

Let us add at this point that theory of real recursive functions is intended to be more analytic in its form than the well-known approach of computable analysis. However, on some levels these theories coincide (see [1]).

We tried to show that our framework is versatile: from a careful and not so complex definition of the (countable) set of recursive functions over the reals we show by means of the toolbox of Analysis that: (a) Laplace transform can be used to quickly obtain useful real recursive functions and to measure their rate of growth, (b) the embedding of Turing machines into continuous time recursive functions is trivial (take a look at the newest definition in [15]), (c) a (limit) hierarchy of real recursive functions exist to classify hardness of functions.

The fact that the set of real recursive functions is countable gives us a possibility to consider decidability questions for these functions. For example it has been proved in [14] that for a real recursive function the problem of its domain is undecidable and the identity of two real recursive functions cannot be determined by any real recursive function.

Let us stop here to study a bit further the mentioned hierarchy of real recursive functions. If $\eta(f)$ counts the smallest rank of the limit operator — the number of nested limits — in every description of a function f, then we can define the following hierarchy of sets:

$$H_i = \{f : \eta(f) \leq i\}.$$

In [12] we established the results that follow.

Proposition 2.1. *The functions* $+$, $\times$, $-$, exp, sin, cos, log *(*inter alia*) are in* H_0*, Kronecker's* δ *function, the function sgn, and Heaviside's* θ *function (*inter alia*) are in* H_1*. Euler's* Γ *function and Riemann's* ζ *function are in* H_1*.*

We can add separation results such as:

Proposition 2.2. $H_0 \neq H_1$ *(since Euler's* Γ *function and Riemann's* ζ *function are in* H_1 *and not in* H_0*.*

About this η-hierarchy (of limits), we may add further topics. We showed in [12] that we can embed the entire arithmetical hierarchy within the limit hierarchy up to some finite level (up to a finite number of limit operations), where the analytic hierarchy starts. The use of limits gives rise to uncomputable functions, e.g., at some level we get the halting problem solved.

This means, *inter alia*, that strong uncompressible numbers like Chaitin's halting probability are found in very precise levels of the limit hierarchy.

Proposition 2.3. *The classical* halting problem *is decidable in some level (*H_3*) of the* η*-hierarchy. Chaitin's* Ω *is a real recursive constant. The Arithmetical Hierarchy is confined to a finite level of the* η*-hierarchy (*H_6*, where the Analytical Hierarchy starts).*

We can prove that the $(H_i)_{i\in\mathbb{N}}$ does not collapse (the full proof can be found in the submitted paper [8]) and contains the whole Arithmetical Hierarchy and the whole Analytical Hierarchy. In fact, Bruno Loff proved in [7] the following most interesting characterization (interesting both for real recursive functions, and for the analytical hierarchy — the later becoming defined without quantifiers and in a single inductive step):

Proposition 2.4. *Real recursive functions are those functions* f *such that the predicate expression* $y = f(x)$ *is in* Δ^1_ω*.*

To these previous aspects, we should add the impact of a further one: (d) in the basis of the limit hierarchy we can still find a set of functions over the real numbers indeed computable by physical means, theoretically by Claude Shannon's General Purpose Analog Computer and practically by the Differential

Analyzer of Vannevar Bush (see [5]). Hence, in H_0 we have truly computable functions in the physical sense (and also in the sense of computable analysis). Is the GPAC the ultimate limit of analog computability? Nobody really knows, but we can add that Rubel improved the GPAC in the 90's building up the conceptual Extended Analog Computer, in a such a way that some limits become physically realizable.

3 Proof methods

In the full version of this paper we prove that if we have a first order differential equation that gives us some function, we can always find an infinite limit that describes the same function, using a numerical approximation which asymptotically behaves in the intended manner. Nonetheless, given a function expressed by an infinite limit, we cannot always find a first order differential equation that results in the same function, because if we could, the η-hierarchy would collapse.

We finish our extended abstract by describing how such a statement can be proved. First, we show that

Proposition 3.1. *There is no universal real recursive function, i.e., there is no real recursive binary scalar function* Ψ *such that, for all* $n \in \mathbb{N}$, $x \in \mathbb{R}$, $\Psi(n, x) = \phi_n(x)$, *where* ϕ_0, ϕ_1, ϕ_2, *... denotes an enumeration of all real recursive functions:* ϕ_n *is the function given by a description coded by* n.

Furthermore, there is no universal real recursive function ψ *which verifies* $\psi(n, x) = \phi_n(x)$ *if* n *codes for a description with the smallest possible rank of the limit operators for the described function.*

Finally, we prove that

Proposition 3.2. *There is a universal real recursive function for each level of the* η*-hierarchy, i.e., for every level* H_n *of the* η*-hierarchy, there is a real recursive binary function* Ψ_n *such that whenever the number of nested limits in a description* e *is less than* n, *we have* $\Psi_n(e, x) = \phi_e(x)$.

These statements taken together prove that the η-hierarchy does not collapse. The function Ψ_n is most probably not in H_n, but it suffices to show that it exists in a higher level of the η-hierarchy.

We conclude that there is no real recursive universal Ψ function, nor even a restriction of Ψ to low-rank codes, but that there are real recursive universal Ψ_n functions for every level of the η-hierarchy. This assures that while we cannot have real recursive characteristics for the problems of domain and identity for every function, we can still have them for every function up to any level of the η-hierarchy. Based on these two statements we prove the main theorem:

Theorem 3.1. *There is no limit to inductive definability of real-valued functions by composition, solving first-order differential equations and infinite limit taking.*

This result makes us feel that our framework can be considered a branch of *Descriptive Set Theory*. For the purpose we recall some words of Yiannis N. Moschovakis (see [10]): *Lebesgue defined the collection of analytically representable functions as the smallest set which contains all constants and projections and which is closed under sums, products and the taking of limits. [...] Today we recognize Lebesgue [1905] [see [6]] as a classical work in the theory of definability. It introduced and studied systematically several natural notions of definable functions and sets and it established the first important hierarchy theorems and structure results for collections of definable objects*. So do we! How close is real recursive function theory to *Descriptive Set Theory*? We do not know, and the answer to this question is an open problem in our research program.

What about connections between *Mathematical Analysis* and *Theory of Computability (and Complexity))* in the other direction? We believe that our most general framework, with infinite limits ([12, 15, 8]), has enough ingredients to allow a good translation of classical computability and classical computational complexity problems into Analysis. We do believe that such translations might be a solution to open problems described in analytic terms: we are much involved in the definition of analog classes P and NP, and to find one good analytic representation of the $P \neq NP$ conjecture (see [13]).

Acknowledgements We would like to thank Fernando Ferreira for having compelled us a bit further towards *Descriptive Set Theory*. Although the title mentions analog computation, our theory is much more about recursive functions over the reals. This change of view is due to Martin Davis who has been systematically disturbed by our proposals: to him we have much to thank.

References

1. O. Bournez, M. Campagnolo, D. Graça, and E. Hainry. The general purpose analog computer and computable analysis are two equivalent paradigms of analog computation. In J.-Y. Cai, S. B. Cooper, and A. Li, editors, *Theory and Applications of Models of Computation (TAMC'06)*, volume 3959 of *Lecture Notes in Computer Science*, pages 631–643, 2006.
2. O. Bournez and E. Hainry. Real recursive functions and real extensions of recursive functions. In Maurice Margenstern, editor, *Machines, Computations and Universality (MCU'2004)*, volume 3354 of *Lecture Notes in Computer Science*, pages 116–127, September 2004.
3. O. Bournez and E. Hainry. Elementarily computable functions over the real numbers and $\mathbb{R}$-sub-recursive functions. *Theoretical Computer Science*, 348(2–3):130–147, December 2005.
4. M. Campagnolo, C. Moore, and J. F. Costa. An analog characterization of the Grzegorczyk hierarchy. *Journal of Complexity*, 18(4):977–1000, December 2002.
5. D. Graça and J. F. Costa. Analog computers and recursive functions over the reals. *Journal of Complexity*, 19(5):644–664, October 2003.
6. H. Lebesgue. Sur les fonctions représentables analytiquement. *J. de Math.*, 1:139–216, 1905.

7. Bruno Loff. A functional characterisation of the analytical hierarchy. In Barry Cooper, Benedikt Löwe, and Andrea Sorbi, editors, *Computability in Europe 2007: Computation and Logic in the Real World*, 2007.
8. Bruno Loff, José Félix Costa, and Jerzy Mycka. Computability on reals, infinite limits and differential equations. *Applied Mathematics and Computation*, 2007. In press.
9. C. Moore. Recursion theory on the reals and continuous-time computation. *Theoretical Computer Science*, 162(1):23–44, August 1996.
10. Y. N. Moschovakis. *Descriptive set theory*. North–Holland, 1980.
11. J. Mycka. μ-recursion and infinite limits. *Theoretical Computer Science*, 302:123–133, June 2003.
12. J. Mycka and J. F. Costa. Real recursive functions and their hierarchy. *Journal of Complexity*, 20(6):835–857, December 2004.
13. J. Mycka and J. F. Costa. The $P \neq NP$ conjecture in the context of real and complex analysis. *Journal of Complexity*, 22(2):287–303, April 2006.
14. J. Mycka and J. F. Costa. Undecidability over continuous-time. *Logic Journal of the IGPL, Oxford University Press*, 14(5):649–658, October 2006.
15. J. Mycka and J. F. Costa. A new conceptual framework for analog computation. *Theoretical Computer Science*, 2007. Accepted for publication.