The Informal Side of Computability: Church-Turing Thesis, in Practice

Luca San Mauro
Scuola Normale Superiore - Pisa, Italy

The aim of this talk is twofold. On the one hand, we propose a philosophical analysis of a somewhat neglected topic in the considerably vast literature concerning Church-Turing thesis (CTT), namely its practical use. On the other hand, we make use of this analysis to enlighten the notion of informal provability.

In doing so, we begin by noticing that, although topics concerning CTT have been extensively investigated (see, for instance, [3] for a recent overview), the way in which CTT is practically used by working mathematicians has received almost no philosophical attention. This fact may strike as surprising. Indeed, an appeal to CTT already appears in a celebrated paper by Post [4], in 1944, as a preliminary justification for the unexpected flavour of informality in which his proofs are given. Then, in 1967, Rogers [6] systematizes this kind of appeals, by defining "proofs by Church's Thesis" those proofs "which rely on informal methods". Clearly, this definition is so inclusive that, according to it, basically every possible construction in Computability ends up having some sort of appeal to CTT (although implicitly).

Despite this sort of ubiquity, a philosophical analysis of the expression "proof by Church's Thesis" is still missing – i.e., an analysis of the gap between the formal presentation of an algorithm, within a certain model of computation, and the ordinary language in which it is commonly formulated. Moreover, sporadic comments strongly deny any relevance to the problem, claiming that "proof by Church's Thesis" just refers to a standard, non significant phenomenon in mathematics (see [1]).

Thus, the primary goal of this talk is to reaffirm the philosophical importance of this practical aspect of CTT. In doing so, we make use of the historical reconstruction sketched above (which we extend to the present days) in order to obtain the following "standard view":

Standard view (of the practical side of CTT):

a) CTT allows us to rely on informal methods;
b) Yet, these methods are in the end just a matter of convenience: informal definitions point towards formal ones, and we could theoretically substitute the former with the latter without any significant loss of information;
c) This operation is analogue to what happens in most parts of mathematics.

It is fair to say that this standard view (SV) has a near universal consensus. This consensus might also explain why the practical side of CTT is so philosophically neglected. Indeed, if supporting SV, one can easily claim that the informal aspects of Computability do collapse onto their formal counterpart. Thus, once justified CTT, there is – philosophically speaking – *nothing more to do*.

In the present talk, we argue against SV, by claiming that it is not adherent to the real practice of Computability. In particular, we show that the success of the Post-Rogers paradigm could have led – at least theoretically – to a messy class of informal descriptions in the definitions of algorithms, since such descriptions do not have to be grounded on any particular model of computation. But historically that was not the case. On the contrary, those descriptions rapidly converged towards an acknowledged standard in the form of their exposition, and their generalizations gave rise to what are called "methods". By carefully examining this notion of method, we then argue that informal algorithms and their formal counterparts differ both logically and conceptually, that is to say, a specific kind of distortion is necessarily embedded whenever the two domains are matched (against item b) in SV).

In supporting this latter claim, our main cases studies are the two following: the construction of a *simple set* and the Friedberg-Muchnik solution to the famous Post's Problem. In both cases, we show that informal constructions are, in practice, not thought as referring to (one of) their formal implementations, but they are rather "structurally" conceived, in the sense that the kind of objects that are constructed are: 1) not extensionally fixed; 2) independent from any specific formal model. In order to clarify these two latter aspects, we borrow the notion of formalism-freeness from [2], and highlight how this feature, in the case of Computability, has its roots in Felix Klein Erlangen Program.

Thus, in conclusion, our general claim is that the informal side of Computability is not fully reducible to the formal one. This latter remark clearly echoes a quite recent and growing literature that aims to clarify the role of informal proofs in mathematics (see, e.g., [5]). Hence, if time permits, in the last part of our talk, we aim to show, by means of some examples, how our analysis might contribute to this research wave.

References

[1] R.L. Epstein, W. Carnielli, *Computability: Computable Functions, Logic, and the Foundations of Mathematics.* Advanced Reasoning Forum, 3rd. edition, 2008.

[2] J. Kennedy, On formalism freeness. *Bulletin of Symbolic Logic* 19, 2013, pp. 351-393.

[3] A. Olszewski, J. Wolenski, R. Janusz, *Church's Thesis After 70 Years.* Ontos, 2007.

[4] E.L. Post, "Recursively enumerable sets of positive integers and their decision problems", *Bulletin of the American Mathematical Society* 50(5), 1944, pp. 284-316.

[5] Y. Rav, "Why do we prove theorems?", *Philosophia Mathematica* 7(1), 1999, pp. 5-41.

[6] H. Rogers, Jr., *Theory of Recursive Functions and Effective Computability*. McGraw-Hill, 1967.