

Finite machines, mental procedures, and modern physics

Rossella Lupacchini

Department of Philosophy, University of Bologna, Bologna, Italy

Abstract. A Turing machine provides a mathematical definition of the natural process of calculating. It rests on trust that a procedure of reason can be reproduced mechanically. Turing's analysis of the concept of mechanical procedure in terms of a finite machine convinced Gödel of the validity of the Church thesis. And yet, Gödel's later concern was that, insofar as Turing's work shows that "mental procedure cannot go beyond mechanical procedures", it would imply the same kind of limitation on human mind. He therefore deems Turing's argument to be inconclusive. The question then arises as to which extent a computing machine operating by finite means could provide an adequate model of human intelligence. It is argued that a rigorous answer to this question can be given by developing Turing's considerations on the nature of mental processes. For Turing such processes are the consequence of physical processes and he seems to be led to the conclusion that quantum mechanics could help to find a more comprehensive explanation of them. (www.actabiomedica.it)

Key words: Finiteness, locality, discreteness

Introduction

The need for a precise and adequate definition of the informal notion of finite procedure emerges in the context of the decision problem and of the generalization of Gödel's incompleteness theorems. Gödel's opening remarks, in his paper of 1931 (1), summarize the result of the work, i.e., the proof of the incompleteness of *Principia Mathematica* and related systems, as follows:

"The development of mathematics toward greater precision has led, as is well known, to the formalization of large tracts of it, so that one can prove any theorem using nothing but a few mechanical rules. The most comprehensive formal systems that have been set up hitherto are the system of *Principia mathematica* (*PM*) on the one hand and the Zermelo-Fraenkel axiom system of set theory (...) on the other. These two systems are so comprehensive that in them all methods of proof used in mathematics are formalized, that is, reduced to a few axioms and rules of inference. One might therefore conjecture that these

axioms and rules of inference are sufficient to decide *any* mathematical question that can at all be formally expressed in these systems. It will be shown below that this is not the case, that on the contrary there are in the two systems mentioned relatively simple problems in the theory of integers that cannot be decided on the basis of the axioms."

Gödel's emphasis is on rules, and the adjective mechanical stands out. The "mechanical rules", as explained in "refer only to the outward structure of the formulas, not to their meaning, so that they can be applied by someone who knew nothing about mathematics, or by a machine." (2).

To disprove the conjecture about the decidability of "any mathematical question", Gödel used an effective technique of "arithmetization" that serves the purpose of constructing, within the formal system, a proposition concerning positive integers which is true, but cannot be proved within the system. The formal system is not capable of producing all true propositions and therefore is incomplete. Moreover, if the negation of the unprovable proposition is added to the

system, a consistent system containing false statements can be formed. Gödel's proof entails the impossibility of asserting consistency for a system that contains a minimal amount of arithmetic without using deductive instruments that cannot be formalized in the system itself: Gödel's second incompleteness theorem.

Although Gödel's incompleteness theorems meant the failure of Hilbert's program, namely to give consistency proofs for mathematics by finitist means (in metamathematics), Gödel's method of proof can be taken as evidence of the strength of the mathematical instruments provided by Hilbert's proof theory. The general approach was explained in Hilbert and Ackermann (3):

"Mathematical logic achieves more than a sharpening of language by a symbolic representation of inferences. Once the logical formalism is fixed, we can expect that a systematic, so-to-speak calculatory treatment of logical formulas is possible that corresponds roughly to the theory of equations in algebra".

According to Herbrand, metamathematics allowed a mathematical treatment of the most general problem of mathematics, the so called decision problem [*Entscheidungsproblem*], "which consists of seeking a method allowing us to recognize with certainty (at the end of a number of operations which can be determined beforehand) whether or not a given proposition is an identity, and if it is to find a proof of the proposition" (4). Herbrand considered the solution of the *Entscheidungsproblem* as another route to establishing consistency. Hilbert and his school viewed the decision problem as a fundamental problem of mathematical logic. Finding a positive solution to the *Entscheidungsproblem* would have meant finding a solution to any mathematical problem using a finite number of operations: it would have been sufficient to formalize, within first-order logic, the branch of mathematics relevant to that question.

But a positive solution to the *Entscheidungsproblem* could also hide the threat that mathematics was trivialized through mechanism. In von Neumann's words (5):

"[...] the undecidability is even the *conditio sine qua non* for the contemporary practice of mathematics, using as it does heuristic methods, to make any

sense. The very day on which the undecidability does not obtain any more, mathematics as we now understand it would cease to exist; it would be replaced by an absolutely mechanical prescription, by means of which anyone could decide the provability or unprovability of any given sentence".

Thus a negative solution to the *Entscheidungsproblem* required to specify which "absolutely mechanical prescription", or finite procedure, would do the job. In a note of 1964, Gödel gives Turing credit for this result (6):

"Turing's work gives an analysis of the concept of 'mechanical procedure' (alias 'algorithm' or 'computation procedure' or 'finite combinatorial procedure'). This concept is shown to be equivalent with that of a Turing machine".

Turing machines

To say that what is "effectively calculable" can be calculated by a machine is to phrase the so-called *Church thesis* in Turing's language, and the machine in question is a *Turing machine*. The Church thesis is a quasi-empirical claim as it provides a precise mathematical definition – such as a Turing machine, Gödel's general recursiveness, or Church's λ -definability – of what can "naturally" be regarded as effectively calculable. It delimits the scope of the concept of calculability. Although those mathematical concepts were proved all logically equivalent, Turing's would reveal a particular perspicuity and a great fruitfulness. Gödel, who hailed Church's definition as "thoroughly unsatisfactory" and suggested "to state a set of axioms which would embody the generally accepted properties of this notion [effective calculability], and to do something on that basis", would claim:

"The most satisfactory way ... [of arriving at such a precise definition of the concept of finite procedure] is that of reducing the concept of finite procedure to that of a *machine with a finite number of parts*, as has been done by the British mathematician Turing" (7)

For Turing effective calculability requires an agent, i.e., a computer. Considering a *human* agent, the goal is then to establish which simple mechanical

operations on symbolic configurations can be carried out in human computing. It follows that boundedness conditions, grounded in limitations of the relevant capacities of a human computer, are to be imposed on computability. If it is assumed that the computation is performed by printing symbols on a potentially infinite tape divided into squares, then there is a lower bound on the size of symbols, to distinguish one symbol from the other, and an upper bound on the number of symbols, to ensure that all symbols (which take part in a computational step) can be observed at a single glance. The states relevant to the computation must be “immediately recognisable”. If the machine is capable of “observing” only one square at the time, then its “steps” must be of two kinds: a change of the observed square to the square one to the left or to the right, a change in the symbol which appears on the observed square.

To know the state of a computation means to know “the sequence of symbols on the tape, which of these are observed by the computer (possibly with a special order), and the state of mind of the computer” (8). This “instantaneous configuration” allows determining which operation is to be executed and what is going to be the next configuration. In this way the capabilities of a human agent of carrying out a calculus can be converted in a finite set of instructions for a logical machine, i.e., a *Turing machine*. Any possible inventiveness should participate in computing is bridled by “operations so elementary that it is not easy to imagine them further divided.” In so far as human computing can be simulated by a finite number of such local operations, a Turing machine is to be understood as *computationally equivalent* to a human agent. *Within these constraints*, a Turing machine calculates like a human being.

As Turing’s method is intended for an application to the *Entscheidungsproblem*, Turing is led to conceive a “universal machine” capable of imitating any Turing machine whatever, and therefore of executing all possible computations. The crucial idea was that when a finite set of instructions which describes any special Turing machine is fed to the universal machine, this simulates the special one. Hence it can execute any computation that any special Turing machine can. Once the Turing machine model fulfils von Neu-

mann’s demand for absolutely mechanical prescription, the *Entscheidungsproblem* is proved by the impossibility of halting a universal Turing machine; moreover, as it provides a “precise and unquestionable adequate definition” of the general concept of formal system, Gödel’s theorem becomes a result concerning the very concept of formal system. The undecidable proposition can now be produced by a method that can be applied with cogency to any formal system. It is worth stressing how Turing’s analysis echoes Hilbert’s words . (9):

“The fundamental idea of my proof theory is none other than to describe the activity of our understanding, to make a protocol of the rules according to which our thinking actually proceeds. Thinking, it so happens, parallels speaking and writing: we form statements and place them one behind another. If any totality of observations and phenomena deserves to be made the object of a serious and thorough investigation, it is this one”.

The question then arises as to whether the very activity of our understanding could be comprised in a few mechanical rules.

Beyond mechanical procedures

As mentioned above, Turing’s reduction of the concept of finite procedure to that of a finite machine convinced Gödel of the validity of the Church thesis. And yet, Gödel’s later concern was that, insofar as Turing’s work shows that “mental procedure cannot go beyond mechanical procedures”, it would imply the same kind of limitation on human mind. He therefore deems Turing’s argument to be inconclusive (10):

“What Turing disregards completely is the fact that *mind, in its use, is not static, but constantly developing*, i.e. that we understand abstract terms more and more precisely as we go on using them, and that more and more abstract terms enter the sphere of our understanding. (...) although at each stage the number and precision of the abstract terms at our disposal may be *finite*, both (and, therefore, also Turing’s number of *distinguishable states of mind*) may *converge towards infinity* in the course of the application of the procedure”..

Is Gödel's critics justified?

For Gödel, our understanding is to involve some sort of purely mental activity beyond the finite memory of the brain. All possible processes of human mind cannot be simulated by a finite state machine. In his Gibbs Lecture (1951), Gödel speculates on some philosophical implications of his incompleteness theorems and focuses on the phenomenon of the inexhaustibility of mathematics. This would be shown very clearly by a theorem which “*makes impossible that someone should set up a certain well-defined system of axioms and rules and consistently make the following assertion about it: All of these axioms and rules I perceive (with mathematical certitude) to be correct, and moreover I believe that they contain all of mathematics*” (7). As a consequence, we are presented with the following disjunction: either mathematics is incompletable (in the sense that its evident axioms can never be comprised in a finite rule), or there exist absolutely unsolvable problems. The first alternative, Gödel asserts, “entails that human mind (even within the realm of pure mathematics) infinitely surpasses the powers of any finite machine.” But does the process of understanding more and more abstract terms necessarily involve the convergence towards infinity of states of mind?

Turing was aware of the difficulties. His *Intelligent Machinery* addresses the issue of Gödel's and other theorems which restrict the powers of discrete states machines. Those mathematical results “have shown that if one tries to use machines for such purposes as determining the truth or falsity of mathematical theorems and one is not willing to tolerate an occasional wrong result, then any given machine will in some cases be unable to give an answer at all. On the other hand the human intelligence seems to be able to find methods of ever-increasing power for dealing with such problems ‘transcending’ the methods available to machines.” (11) But this argument, in Turing's opinion, rests essentially on the condition that the machine must not make mistakes, and yet this condition is not a requirement for intelligence. Moreover, “although it is established that there are limitations to the powers of any particular machine, it has only been stated, without any sort of proof, that no such limitations apply to the human intellect.” (12)

Turing was always intrigued by the nature of mental processes, but (differently from Gödel) he believed that, in the final analysis, they were the consequence of physical processes. His conceptual analysis rests on the identification of the activity of a human computer with that of the finite state machine. Hodges calls attention to Turing's idea that “mental processes are correctly described in the logical model independently of the particular physical embodiment, and so can be embodied in a physical form other than the brain.” (13) The possibility of realizing the logical model gives rise to a process of feed-back, to the search for those logical functions typical of intelligence that might be implemented in an actual machine.

In a *Lecture to the London Mathematical Society* (1947), Turing discussed possible physical implementations of his universal machine model. What amounts to a machine capable of showing intelligence? Here the question of memory is crucial, and the requirement for a large storage capacity emerges out of the demand for a machine capable of anything more than rather trivial operations.

“Some years ago [...] I considered a type of machine which had a central mechanism, and an infinite memory which was contained on an infinite tape. This type of machine appeared to be sufficiently general. [...] It was essential in these theoretical arguments that the memory should be infinite. It can easily be shown that otherwise the machine can only execute periodic operations. Machines such as the ACE may be regarded as practical versions of this same type of machine.” (14)

But can computing machines only carry out the processes that they are instructed to do? Let us suppose, Turing suggests, to have set up a machine with certain initial instruction tables, so constructed to be capable of altering its own instructions out of all recognition and, as a result of this, of executing its tasks in a much more efficient manner. When this happens are we not obliged to regard the machine as showing intelligence? “As soon as one can provide a reasonably large memory capacity it should be possible to begin to experiment on these lines. [...] What we want is a machine that can learn from experience. The possibility of letting the machine alter its own instructions provides the mechanism for this, but this of course does not get us very far.” (14)

The issue of machine learning can be formulated as follows. Different (Turing) machines can be regarded as different proof-finding (Turing) machines. The question then arises as to whether the procedure to be applied for choosing between different machines is computable. Would it be considered by Turing as a matter of “intuition”? In line with Copeland, a mathematician can be viewed as a proof-finding machine. “When intuition supplies the mathematician with new means of proof, then he becomes a different proof-finding machine, capable of a larger set of proofs.” (15) The search for new methods of proof is motivated by problems to solve, and yet it seems to involve some *random* element. How can a machine perform this search? In so far as this is a process of learning from experience, Turing’s answer would be that a machine can do it by altering its own instructions. Does this imply a transformation of a proof-finding machine into a different one? According to Copeland, what is at issue here is a “partially random” learning machine which “emulates the ‘activity of the intuition’ in its walk through the space of proof-finding Turing machines” (15). Since it is determined by both experience and intuition, the trajectory of the learning machine through this space might be uncomputable.

The *deterministic* character of Turing machine computations satisfies Gödel’s requirement for a limited freedom in the activity of the mathematician – “If anything like creation exists at all in mathematics, then what any theorem does is exactly to restrict the freedom of creation” (7) – however, while the (universal) Turing machine, as derivative of the human model – “A man provided with paper, pencil, and rubber, and subjected to strict discipline, is in effect a universal machine.” (11) – is anchored in space and time, for Gödel mathematical experience involves an objective reality out of space and time.

It is correct that a mathematical proposition says nothing about the physical or psychical reality existing in space and time, because it is true already owing to the meaning of the terms occurring in it, irrespectively of the world of real things. What is wrong, however, is that the meaning of the terms (that is, the concepts they denote) is asserted to be something man-made and consisting merely in semantical conventions. The

truth, I believe, is that these concepts form an objective reality of their own, which we cannot create or change, but *only perceive and describe*. (7)

What does such an objective reality imply? To state that the infinite mathematical concepts form an objective reality does it not imply to cancel the difference between *finite* and “infinite” intellect? If knowledge comes in with perception, beyond the limits of experience, it has no meaning because it is detached from experience. Here is the reason for the critical analysis to negate the legitimacy to this form of passage to limit. As lucidly emphasized by Cassirer, if a finite intellect is “limited”, this limitation holds not only a negative mark, but a positive one as well. It does not show an accidental and exterior limit of the intellect, it is rather a necessary condition for its activity and fruitfulness. It is not licit to conceive a limit as a simple obstacle to avoid: it rather delimits the only domain in which our thought and our knowing can realize themselves, the domain in which they find their concrete meaning (16). Thus the restriction of the intellect to the conditions and limits set by experience is its only possibility of realization.

In this perspective, the fault of Gödel’s views lies with getting over the empirical constraints of knowing, while the significance of Turing’s conceptual analysis emerges with a particular clearness.

Computability and physical principles

Recently the issue of computability has been extensively discussed by Sieg who recognizes the genuinely distinctive character of Turing’s analysis in the move from arithmetically motivated calculations to general symbolic processes that underlie them and have to be carried out *programmatically* by human beings [...]; it was the normative demand of radical intersubjectivity between humans that motivated the step from axiomatic to formal systems. It is, for this reason, that Turing brings in human computers in a crucial way and exploits the limitations of their ‘processing capacities’, when proceeding mechanically (17).

The issue of how computability impinges on the relevant capabilities and limitations of the computer is generalized by Gandy (18) to the case of a discrete

mechanical device. What remains of Turing's analysis, when we purge the computer of the "human" aspect? The answer is given by fixing principles for mechanisms that are as general as possible and by formulating a *mechanical* version of the Turing thesis in terms of a "Gandy machine". By contrast to a Turing machine, whose restrictions are set by human limitations, a Gandy machine has to comply with physical limitations, that is, a lower bound on the size of "atomic components" and an upper bound on the speed of signal propagation, that is the speed of light. A "*locality condition*" set by *relativity theory*.

A Gandy machine is a deterministic mechanical device, which operates in parallel on an arbitrary number of bounded parts. Because of the parallelism involved, here the machine has to recognize patterns in a given state and act on them locally. Yet the states of the computation remain "immediately recognisable". Improving the "(resolution) power" of the computer does not change the scope of computability. The Gandy thesis does not question the Church thesis, but it does corroborate its significance from the point of view of conceptual analysis. A Gandy machine does not come to terms with the sensory limitations of a human being but rather with physical constraints. In line with Turing, the restrictions imposed on the operations performed by computers rest upon the computers' capacities. For both, this means putting boundedness conditions on the (set of) symbols a computer can recognize. Thus, it can be argued that the strength of this conceptual argument lies in the fact that an intimate and "inescapable" connection is established between computability and "measurability".

If computation is conceived as an "activity" which can be simulated by a Turing machine, which characteristic features of this activity make the simulation possible? Turing focused on human computing activity; Gandy on discrete deterministic mechanical processes. A further step has been taken by David Deutsch in 1985. Interpreting what is effectively calculable as calculable by a quantum physical system allows him to design a "universal quantum computer". The description of a quantum computer is derived from a Turing machine, but using quantum theory to define the operations carried out by the computer. Hence it can exploit a multiplicity of parallel computational paths

in superposition as well as quantum interference to amplify the probability of correct outcomes of computations.

If Turing machines calculate like humans, then quantum Turing machines mimic physical systems that calculate. What made this shift possible was the preliminary assertion that "Every finitely realizable physical system can be perfectly simulated by a universal Turing computing machine operating by finite means." (19) Following Deutsch a computing machine operates by finite means if: (i) only a finite subsystem is in motion during anyone step; (ii) the motion depends only on the state of a finite subsystem; (iii) the rules that specify the motion can be given finitely in the mathematical sense (for example by an integer). "Turing machines", Deutsch asserts, "satisfy these conditions, and so does the universal quantum computer." In line with Turing and Gandy, once more finiteness conditions set the scope of computability. Yet in quantum computing, those finiteness conditions mirror/rest on/reflect physical constraints.

The description of a *quantum Turing machine* is derived from a Turing machine, but using quantum principles to define the operations carried out by the computing agent, which is now a *physical system*. It follows that the state of each single unit, called *qubit*, is a "coherent superposition" of two classical logical states (0 and 1). When a computational state can be reached through several alternative paths, its probability is the squared modulus of the sum of all the "probability amplitudes" for the constituent paths. Quantum computation arises from the possibility of exploiting a multiplicity of parallel computational paths in superposition as well as quantum interference to amplify the probability of correct outcomes of computations. Since quantum mechanics describes a state evolution by means of a unitary operator, any quantum computing operation is a unitary transformation on qubits. Quantum interference allows a quantum Turing machine to act on coherent superpositions of a given state and evolve them *via* unitary operations into other superpositions, from which the next state results with a certain probability.

It is worth noticing two characteristic features of quantum physics from which additional computational power might flow. First, unlike probabilities, the

probability amplitudes involved in quantum mechanics do not necessarily introduce uncertainty. Since the probability amplitudes are complex numbers, they may cancel each other and produce *destructive* interference, or enhance each other and produce *constructive* interference. Secondly, as a quantum state, and therefore a qubit, consists of *all* the linear combinations of its basis states, constrained only by the normalization condition, it has to contain an infinite amount of information. As a consequence of this all, it can be tempting to conjecture that “instead of being limited to shuffling a finite collection of [classical computational] states through permutation, one can act on qubits with a continuous collection of unitary transformations.” (20) This appearance of infinity and continuity seems to question the requirement for finite means. But quantum principles “protect” Turing computability.

If one shares Turing’s concern for effectively distinguishable states, the question at issue here is how to discern quantum (computational) states. As Dirac (21) reminds us, the superposition that occurs in quantum mechanics is of an essentially different nature from any occurring in the classical theory. The quantum superposition principle demands indeterminacy in the results of observations in order to be capable of a sensible physical interpretation. In classical physics, each constituent part of matter is itself made up of smaller parts, in terms of which its behaviour is to be explained, and there is no end in this procedure. By contrast, in quantum theory, it is necessary to give an absolute meaning to size:

“In order to give an absolute meaning to size (...) we have to assume that there is a limit to the finiteness of our powers of observation and the smallness of the accompanying disturbance - a limit which is inherent in the nature of things and can never be surpassed by improved technique or increased skill on the part of the observer” (21).

In spite of the profound differences, Turing computability and quantum physics present us with two similar fundamental questions: how would it be possible to encode the input and read the output of a computation or to make measurements and predictions concerning a physical quantity without assuming a limit to the fineness of our means of observation? How

could an adequate description of any “effective” procedure dispense with the *medium* of the agent (computer or observer) capable of carrying out the operations involved?

Open questions

As refers his biographer (22), Turing himself was puzzled by the nature of quantum indeterminacy and by the *reduction* rules which apply to quantum measurement. He was not so much interested in the unitary evolution of a quantum system as to the “unpredictable” process of reduction which takes place in quantum measurement. How could a continuum of states be reduced to a discrete spectrum of observable values? Could quantum physics, with its continuous wave function and indeterminacy principle, be compatible with Turing computability? Turing seemed to focus on the fact that subjecting a quantum system to continual observation has the effect of preventing its dynamical evolution (motion), and conjectured a “non-linear” description in quantum mechanics beside “linear” predictions. Could he have seen a connection between the problem of the linearity in quantum mechanics and the question of the computability of the intelligence? To convert a brain or a machine into a universal machine, in Turing’s opinion, is the extreme form of *discipline*, what is required in addition to produce intelligence we call *initiative*. “Our task is to discover the nature of this residue as it occurs in man, and to try and copy it in machines.” (11) Might quantum physics throw some light on the nature of this residue?

References

1. Gödel K. On formally undecidable propositions of Principia mathematica and related systems I. (1931). In Feferman S et al., eds: *Collected Works*, vol 1. Oxford U.P., Oxford, 1990-95: 5-38.
2. Gödel K. The present situation in the foundations of mathematics. (1933). In *CW* (1), vol 3: 45-53.
3. Hilbert D, Ackermann W. *Grundzüge der theoretischen Logik*. Berlin, Springer-Verlag, 1928.
4. Herbrand J. Les bases de la logique hilbertienne. *Revue de métaphysique et de morale* 1930; 37: 243-55.

5. von Neumann J. Zur Hilbertschen Beweisstheorie. *Mathematische Zeitschrift* 1927; 26: 1-46.
6. Gödel K. Postscript to "On undecidable propositions of formal mathematical systems". (1964). In CW (1), vol. 1: 369-371.
7. Gödel, K. Some basic theorems on the foundations of mathematics and their implications. (1951) In CW (1), vol. 3: 304-323.
8. Turing AM: On computable numbers, with an application to the Entscheidungsproblem. *Proc London Math Soc II*; 1936-37: 42.
9. Hilbert D. The foundations of mathematics. (1927) In van Heijenoort J, ed: From Frege to Gödel. Harvard U. P., Cambridge Mass., 1967: 464-79.
10. Gödel K. Some remarks on the undecidability results. (1972) In CW (1), vol. 2: 305-306.
11. Turing AM. Intelligent machinery. (1948) In Ince DC, ed: Collected Works of Alan Turing: Mechanical Intelligence. North-Holland, Amsterdam, 1992: 107-131.
12. Turing AM. Computing machinery and intelligence. (1950) In CW (11): 133-160.
13. Hodges A. Alan Turing and the Turing machine. In Herken R, ed: The Universal Turing Machine. A Half-Century Survey. Oxford U.P., Oxford, 1988: 3-15.
14. Turing AM. Lecture to the London Mathematical Society on 20 February 1947. In CW (11): 86-105.
15. Copeland BJ: Hypercomputation: philosophical issues. *Theoretical Comp Science* 2004; 317: 251-67.
16. Cassirer E. Determinismus und Indeterminismus in der modernen Physik. Göteborg; Elanders Boktryckeri Aktiebolag, 1937.
17. Sieg W. Gödel on computability. Pittsburgh; Mimeo 2005.
18. Gandy R. Church's thesis and principles for mechanisms. In Barwise J et al., eds: The Kleene Symposium. North-Holland, Amsterdam, 1980: 123-48.
19. Deutsch D. Quantum theory, the Church-Turing principle and the universal quantum computer. *Proc R Soc London* 1985; 400: 97-117.
20. Mermin D. From Cbits to Qbits. Teaching computer scientists quantum mechanics. *Am J Phys* 2003; 71: 23-30.
21. Dirac P. The Principles of Quantum Mechanics. Oxford; Clarendon Press, 1930.
22. Hodges A. What would Alan Turing have done after 1954? In Teuscher C, ed: Alan Turing: Life and Legacy of a Great Thinker. Springer, Berlin, 2004: 43-58.

Correspondence: Rossella Lupacchini
 Dipartimento di Filosofia
 Università di Bologna
 Via Zamboni, 38
 I-40126 Bologna
 Tel +39 051 2098330
 Fax +39 051 2098355
 E-mail: rossella.lupacchini@unibo.it