PENROSE'S VIEW ON THE FOUNDATIONS OF MATHEMATICS, THE COMPUTABILITY OF HUMAN CONSCIOUSNESS AND GÖDEL'S THEOREM

DANIEL HEREDIA
Universidad de Sevilla

1. INTRODUCTION

Roger Penrose is considered one of the most important physics-mathematicians today, and part of this recognition has to do with the interdisciplinary nature of the ideas he expounds. In the field of philosophy, for example, his proposals have achieved great notoriety. There are many philosophical debates in which Penrose enters and in each of them our author leaves his personal stamp, which in most cases is not without controversy.

In this work we will see how Penrose argues that the debate on the computability of human consciousness is ultimately related to the debate on the foundations of mathematics.

Section 2 is focused on the arguments that Penrose offers to defend that both debates are related. In this section the figure of Turing will be highlighted, who is, according to Penrose, in some way the precursor of what he himself tries to defend. Section 3 contains a main theme in Penrose's approaches in the debate on the possibility of computability of human consciousness, that is, the acceptance of Gödel's theorem as a firm argument against this possibility. This section is divided, in turn, into two subsections. A rough explanation and context of Gödel's theorem will be given in 3.1. In 3.2 we will have the opportunity to see how Penrose adopts Gödel's theorem as an argument against the computability of human consciousness. In section 4 we will review the main criticisms that Penrose's proposal has received, from the first ones he received to other more current ones.

2. DISCUSSION: THE COMPUTABILITY OF CONSCIOUSNESS AND ITS RELATION TO THE FOUNDATIONS OF MATHEMATICS

The main reason why Penrose argues that the problem of the computability of consciousness has its origin in the foundations of mathematics has a historical basis.

When formalists and intuitionists engaged in the debate about the foundations of mathematics, answers arose with respect to different problems. Specifically, in 1900 David Hilbert proposed at what would become the second International Congress of Mathematicians in Paris his series of 23 problems that, according to him, would occupy future mathematical research. Of this series, the tenth problem would turn out to be the key to the approach to the computability of consciousness as we know it today. The tenth problem proposes "to find an algorithm that determines whether a given polynomial diophantine equation with integer coefficients has an integer solution". However, this tenth problem would only partially lead to the discussion of the computability of consciousness. It would not be until 1928 when the posing of another problem (which followed the same idea as the tenth in Hilbert's series) paved the way for the problem of computability of consciousness to enter the scene. On that date, Hilbert himself together with his colleague Wilhelm Ackermann proposed in Bologna what is known as the *Entscheidungsproblem*. What is posed in this problem is the existence of a mechanical-algorithmic procedure that accounts for certain mathematical problems[43] related to formal systems. The solution was found relatively soon (specifically in 1936) by Alonzo Church and Alan

[43] The concrete mathematical problem is to know whether it is possible that such an algorithm could decide whether the rules of a formal system can be proved.

Turing[44], being the answer to the problem negative (such an algorithm does not exist).

Far from making a detailed analysis of the solutions proposed by Church and Turing, we will see some of the characteristics of both, although with more emphasis on Turing's particular one. The reason for this is that Turing's contributes to the debate on the computability of consciousness more directly than Church's, even though both say the same thing. In the following, we will see why.

Church's solution consisted in the realization of an abstract scheme that finally gave the negative answer to the halting problem. This scheme is known as the lambda calculus. This solution highlights the mathematical nature of the notion of computability. But the notion that Church handles, however, has little to do with computing machines, at least in the first instance. The lambda calculus is so abstract that its application to anything *beyond* mathematics is difficult to contemplate.

Not so with Turing's solution. He posed the problem in terms of a hypothetical machine, which would later bear his name. The Turing machine talks about in his solution to the problem is an abstract machine. Nevertheless, being abstract did not prevent this machine from becoming the driving force behind the creation of computers as we know them today. The reason why Turing's machine has been so important for the development of today's computer machines lies precisely in the proposal of how this machine should work. The characteristics of its operation and its composition are as follows[45]:

> - Tape: Although modern computers use a random access device with finite capacity, the memory of the Turing machine is infinite.
>
> - Read/write head: The read/write head at any time points to a symbol on the tape […] The read/write head reads and writes one symbol at a time from the tape. After reading and writing it moves left, right or stays

[44] The answers of both were given independently. But it is important to recognize the possible influence that Church's work had on Turing's answer (to the extent, for example, that the former was the thesis director of the latter), although not in his approach, but in his line of research.

[45] There are several versions of the Turing machine, but the one presented in the quote contains the features common to all these versions.

> in place. Reading, writing and moving are all performed under instructions from the controller.
>
> - Controller: The controller is the theoretical counterpart of the central processing unit (CPU) in modern computers. It is a finite-state automaton, a machine that has a predetermined finite number of states and moves from one state to another based on input. At any time it can be in one of these states
>
> [...] For each read of a symbol, the controller writes a character, defines the next position of the read/write head and changes the state [...]. For each problem, we must define the corresponding table (Forouzan, 2003: 321-322).

Despite the simplicity of the machine, it has the capacity to perform a large number of procedures. However, what Turing demonstrates with the hypothesis of the existence of this machine is precisely that the way of proceeding (mechanical-algorithmic) cannot account for the halting problem. Even obtaining a machine with unlimited capacity in the handling of algorithms, it cannot solve the *Entscheidungsproblem* proposed by Hilbert and Ackermann. Therefore, the Turing machine reveals the full potential of algorithms, but, at the same time, it also warns of their limitations.

It is almost paradoxical that a hypothesis whose conclusion is the limitation of algorithmic systems should contribute to the [philosophical] approach to the possibility of an artificial intelligence. But in reality the paradox does not occur, since Turing's research interests pointed in another direction.

One of the topics that marked Turing´s path research was (as Penrose believes) that pertaining to the foundations of mathematics. What was the nexus that united two subjects, in principle, so distant? Ivor Grattan-Guinness gives the influence of one of Turing's mentors, Max Newman, a fundamental weight in the role of said nexus.

Grattan-Guinness base his argument on the fact that Newman was an expert in the debate on the foundations of mathematics. Not only that, but he also made a notable contribution to that debate. Newman focused his research on topology, being a pioneer in Great Britain to investigate in this branch of mathematics; and also in logic, having a great importance Russell's logicism (Grattan-Guinness, 2017: 441). His interest in these fields was not the result of chance. Newman belonged to the doctrine of mathematicians who thought of the need to solve the rift between logic and mathematics and his research would be occupied in this task.

One event that was definitive in terms of Newman's influence on Turing's research approach was a course that Newman taught at Cambridge in 1935. Such courses had as their main theme the foundations of mathematics, the perspective of the Brouwerian intuitionists being of special importance. Within the topic of the foundations of mathematics, different problems derived from this debate were also explored, such as the decision problem and the contributions made by Gödel a few years earlier. Although Turing did not mention the importance of such courses, it is presumed very likely that the contact between him and Newman was for the development of the 1936 article about computability (Grattan-Guinness, 2017: 439).

So far, an important part of the debate on the foundations of mathematics has been relegated to the background, and this is Gödel's contribution to it. Such a contribution is the incompleteness theorem, which will be of great importance in Penrose's thought.

We have then that Kurt Gödel is, together with Turing and Church, one of the three pillars regarding the search for the meaning and limits of computability (Copeland, 2017: 57). If the role of these three personalities has sometimes not been given the appreciation it deserves, it is because their work was focused on the abstract and not the practical nature of the term computability. This has also taken its toll in making recognizable the substantial connection between computability and the foundations of mathematics. The differentiation of the practical and abstract plane of computability has been the obstacle that has prevented such a connection from being manifest.

The fact is that the role of the three is undeniable. However, what they defended led to different paths. Church and Turing's proposal served as a parapet for those who defend the possibility of the computability of human consciousness, while Gödel's served as a parapet for the detractors of this position. As we saw above, Penrose is one of those thinkers who use Gödel's ideas to debate the arguments of the defenders of the possibility to create an artificial intelligence. Let us see how he does it.

3. MORE ASPECTS OF THE DISCUSSION

3.1. A BRIEF LOOK TO GÖDEL'S THEOREM

It is useful to know in what context Gödel´s theorem arose, to see in which way Penrose adopts it to his arguments against the posibility of the computability of the mind and human consciousness. Kurt Gödel, at the age of 25, presented the theorems that would bear his name in response to the trend in mathematics at the time:

> As is well known, the progress of mathematics toward ever greater accuracy has led to the formalization of large parts of it, so that deductions can be carried out according to a few mechanical rules. The most extensive formal systems constructed so far are the *Principia Mathematica* (PM) system and the axiomatic set theory of Zermelo-Fraenkel (further developed by J. von Neumann) (Gödel, 2006: 53).

Mathematics developed especially by Hilbert, Russell and Whitehead (for Penrose[46], they are formalists), were translated into formal systems (logical systems being of capital importance), and every mathematical problem could be solved by means of them. This had as a consequence the conception of mathematics as a mechanical knowledge (Detlefsen, 1996: 80). This, however, did not convince the young Gödel:

> These two systems are so broad that all the methods used today in mathematics can be formalized in them, that is, they can be reduced to a few axioms and rules of inference. It is therefore natural to conjecture that these axioms and rules suffice to decide all mathematical questions that can be formulated in these systems. In what follows it is shown that this is not so, but that, on the contrary, in both systems there are relatively simple problems of natural number theory that cannot be decided with their axioms (and rules) (Gödel, 2006: 53-54).

What was proposed with this theorem, therefore, was to throw away the hope of being able to explain mathematics by means of a certain number

[46] Although commonly those who are known as formalists are the followers of Hilbert, while the followers of Russell and Whitehead are called logicists.

of axioms and rules. That is to say, to return to the question of the foundations of mathematics. Something he finally succeeded in doing.

Although the explanation of the Gödelian theorem will be panoramic, it is necessary to clarify some concepts. The purpose of the theorem is that a proposition belonging to a concrete formal system can declare itself as undecidable. This would have as a consequence the proof of its indeducibility. And not only that, but it could also be translatable to any formal system.

First of all, let's see what a proposition is. A proposition is defined as those statements that generally respond to a truth value. The truth value of a proposition is defined in a basic way, being true when the proposition is true and false when it is false (Russell, Whitehead, 1997: 7). This concept, therefore, is of capital importance within logic. The second important concept of purpose is deducibility, which is understood as the ability to account for logical consistency[47], in syntactic terms, i.e. structure. And the third concept, which does not appear in the purpose but does carry weight, is that of recursion. This term corresponds to the capacity of a procedure to define itself. Among the recursive processes, those that interested Gödel for his theorem were the primitive recursive functions, which are those that define themselves when their main operations are composed of recursion and composition of functions. It is appropriate that we retain this idea of self-explanatory ability, because it is of fundamental importance.

Returning again to the purpose, we have that it revolved around the idea that a proposition could account for its undecidability. That is, the theorem aims to make manifest to what extent a formal system can account for itself, or rather, to what extent it cannot account for itself. In order to reach this result, two steps were required: i) the construction of such a proposition and ii) the proof of the undecidable character of the

[47] This is a property of a formal system, which consists, roughly speaking, in understanding as impossible the acceptance of a concrete system and its contradiction at the same time.

proposition[48]. The fact is that Gödel ends up successfully accomplishing these two steps[49] and thus achieves his purpose.

Penrose explains through a few simple logical steps the implications of this theorem as follows:

> We have numbered all propositional functions which depend on a single variable, so the one we have just written down must have been assigned a number. Let us write this number as *k*. Our propositional function is the *k*th one in the list. Thus
>
> $\neg\exists x\ [\prod x \text{ proves } Pw\ (w)] = Pk\ (k)$
>
> We will now examine this function for the particular w value: w =k. We have:
>
> $\neg\exists x\ [\prod x \text{ proves } Pk\ (k)] = Pk\ (k)$.
>
> The specific proposition Pk(k) is a perfectly well-defined (syntactically correct) arithmetical statement. Does it have a proof within our formal system? Does its negation ¬ P*k*(*k*) have a proof? The answer to both these questions must be “no”. We can see this by examining the *meaning* underlying the Gödel procedure. Although P*k*(*k*) is just an arithmetical proposition, we have constructed it so that it asserts what has been written on the left-hand side: 'there is no proof, within the system, of the proposition P*k*(*k*). If we have been careful in laying down our axioms and rules of procedure, and assuming that we have done our numbering right, then there cannot be any proof of this P*k*(*k*) within the system. For if there were such a proof, then the meaning of the statement that P*k*(*k*) actually asserts, namely that there is *no* proof, would be false, so P*k*(*k*) would have to be false as an arithmetical proposition. Our formal system should not be, so badly constructed that it actually allows false propositions to be proved! Thus, it must be the case that there is in fact *no* proof of P*k*(*k*). But this is precisely what P*k*(*k*) is trying to tell us. What P*k*(*k*) asserts must therefore be a *true* statement, so P*k*(*k*) must be true as an arithmetical proposition. We have found a *true* proposition which has *no proof within the system*! (Penrose, 1991: 146-147; italics in the original).

The statement of the proposition P*k*(*k*), which is within a previously established formal system is true. But such a truth value cannot be proved within the system, at least without falling into a contradiction. We

[48] This division into two concrete steps is proposed by Jean Ladrière to understand the theorem in a more schematic way. For such an analysis of the Gödelian theorem see his work (Ladrière, 1969).

[49] For a detailed explanation of the theorem see Gödel (2006: 57-87).

have then that recursion is not a capability that belongs to formal systems. When formal systems try to give an explanation from themselves without falling into contradiction they can only manage to go round and round without arriving at any concrete answer. This situation is what is known in logic as a *cycle* or *vicious circle*, which is a point of deadlock from which it is impossible to get out.

An algorithm has the same structure and functionality as a formal system[50]. If formal systems are not able to account for themselves, algorithms, for their part, will not be able to do so either. In fact, this is what happens! Human beings have the capacity to be able to account for themselves or, at least, the exercise of reflection does not lead them to a situation like the vicious circle to which formal systems are sometimes condemned. This is basically the way in which Penrose understands that Gödel's theorem can be decisive against the possibility of an artificial intelligence. But let us look at it more closely.

3.2. How Penrose adopts Gödel's theorem to his ideas

Penrose is not the first to highlight the potential of Gödel's theorem as a basis for arguing against understanding mental faculties as mere computations. The role of pioneer belongs to John Lucas, who in 1961 used Gödel's theorem as a support for mentalism, in its confrontation with physicalism. Penrose himself recognizes such influence, although he also claims a particular contribution, which, he considers, overcomes the difficulties that Lucas' argument had to go through (Penrose, 2012: 65).

In any case, at first[51], Penrose treated Gödel's theorem succinctly. It would be later[52] when he considered that this theorem has a wider scope than he first supposed. Penrose will use both the implications of the theorem itself and what it implied in its context.

[50] This is not a personal consideration. It is usually accepted in this way, i.e. see Penrose (2012: 108). In any case, it is advisable to be careful with this kind of statements, as Juliette Kennedy points out in (2017: 71).

[51] In *The Emperor´s new mind* (henceforth ENM).

[52] In *The Shadows of the mind* (henceforth SOTM).

Gödel's theorem arose to make it clear that mathematics is beyond formal systems. Penrose makes the same claim, but referring to human consciousness and computational processes.

One of the bases on which Penrose sustains the adoption of Gödel's theorem for his arguments has to do with the limits of computational processes[53].

Penrose has no problem in admitting the ability of machines to solve problems of a complexity that may even be impossible for human beings. His defense is based on the fact that it is impossible for a machine to acquire a consciousness *equal* to that of human beings. In fact, he thinks (Penrose, 2012: 60) that it is in the problems that require a solution through common (human) sense that machines are far removed from human beings.

Therefore, the "less brilliant" part of human consciousness would be that which would most deeply engage machines in their attempt to catch up with our species. Penrose is aware that his position has not infrequently been interpreted in the opposite way and therefore clarifies:

> [...] I am claiming that "understanding" involves the same kind of non-computational process, whether it lies in a genuine mathematical perception, say of the infinitude of natural numbers, or merely in perceiving that an oblong-shaped object can be used to prop open a window, or in understanding how an animal may be secured or released by a few selected motions of a bit of rope, or in comprehending the meanings of the words "happiness", "fighting", or "tomorrow" (Penrose, 2012: 69).

Now, what does this have to do with Gödel's theorem? Penrose argues that this way of *skipping* computational rules is what directly relates the Gödelian theorem to the non-computability of human consciousness. Recall that this theorem tells us, precisely, that formal systems cannot account for themselves, since they can have additional rules that are not contained in them, but outside. Only from a non-computational procedure could such additional rules be accounted for. We know that human beings have non-computational procedures, but to propose that a

[53] For a recent study of the reasons for Penrose's adoption of Gödel's theorem, see Heredia (2019: 168-177).

machine also possesses them is a flagrant contradiction because machines owe their behavior to computational processes!

Despite the connection between Penrose's arguments and Gödel's theorem, this does not imply that the thinking of both authors is in consonance. Let us recall that Gödel did not rule out that mathematical thought responds to a subtle algorithm:

> […] (Gödel) found himself seemingly driven in the mystical direction […] that the mind cannot be explained at all in terms of the science of the physical world (Penrose, 2012: 143-144).

Does this mean that Penrose is inappropriately forcing the implications of Gödel's theorem? In my opinion, I think he is not. And as a general rule this is not an aspect that he is usually reproached for. In the following section we will see in what way Penrose's position is criticized and to what extent these criticisms have, in my opinion, weight or not.

4. CONCLUSIONS

That Gödel's theorem can be a useful tool to argue against those who defend Artificial Intelligence is something that does not seem to please everyone. This, however, is not the direction I take. I think that the idea of confronting the concept of reflection and recursion is a great success and, I believe, that it is a strong philosophical argument. Penrose says with respect to this, "Reflection principles provide the very antithesis of formalist reasoning. If one is careful, they enable one to leap outside the rigid confinements of any formal system to obtain new mathematical insights that did not seem to be available before" (Penrose, 1991: 151). Reflection is a feature that seems beyond the reach of machines. Although this argument makes it possible to cover issues that may at first seem distant, this does not prevent it from being subject to criticism.

Some of the criticisms we will see below are directed both to the use of Gödel's theorem by John Lucas and to that carried out by Penrose. On the other hand, indirect criticisms of both will also be included.

The first of the criticisms is that one carried out by Russell and Norvig in their joint work (2004). This particular critique also takes into account Penrose's perspective, although it is recognizable that the interest of the critique is centered on that of Lucas[54]. Russell and Norvig argue that the adoption of Gödel's theorem is open to criticism on three different points.

The first highlights how Lucas (Penrose also does so) understands computational processes to be those carried out by Turing machines. Thus, to speak of computers and Turing machines is to speak of the same thing. While this is easily acceptable, it is no less true that this should not be understood in an absolute way, since Turing machines are infinite, while computers are finite. This infinite character of Turing machines implies that they are not subject to what Gödel's theorem dictates. Not only that, but this could be transferred to any computer (Russell & Norvig, 2004: 1078).

This part of the criticism does not carry the weight it is intended to give. It is raised as if in employing the adoption of Gödel's theorem all the features of the Turing machine had not been taken into account. Penrose is quite clear on this point. In fact, he defends that his point of view conflicts with the thesis of Turing and his machine, while with Church's thesis (more abstract) it need not necessarily clash head-on (Penrose, 2012: 35). Penrose places Turing's machine and Gödel's theorem in the same scenario, because he argues that Turing himself would have accepted it (Penrose, 2012: 35).

The second point is that the limitations of the Gödelian theorem are neither so dramatic for machines nor so definitive as an argument. To make this idea manifest, Russell and Norvig argue that we humans could find ourselves in a similar predicament to that faced by formal (and computational) systems. They put it this way:

> […] Consider the sentence
>
> J. R. Lucas cannot consistently assert that this sentence is true.

[54] As a comment I would like to say that I find it striking that being this work closer to Penrose's (being SOTM from 1994 and the work of Russell and Norvig from 1995, with later editions in 2003 and 2009) the critics continue to focus mostly on the work of John Lucas, which is 30 years older.

> If Lucas asserted this sentence, then he would be contradicting himself, so therefore Lucas cannot consistently assert it, and hence it must be true. We have thus demonstrated that there is a sentence that Lucas cannot consistently assert while other people (and machines) can. But that does not make us think less of Lucas (Russell & Norvig, 2004: 1078-1079).

Of course, the situation described does not change the idea with respect to Lucas, but it is also necessary to understand that a real situation is not being described. In logical terms it is true, the human being would suffer the same fate as a formal system or a computer, i.e., he would be cornered by the sentence. But the difference lies in the fact that the human being can account for such a logical labyrinth, while a formal system or any machine (however powerful it may be), cannot. If our idea about Lucas does not change, it is precisely because it should not!

Later, they argue that if the inferiority that humans have with respect to machines in terms of rapid calculation is not taken into account to discredit human intelligence, why is the opposite the case with machines and their limitations (Russell & Norvig, 2004: 1079).

This again is in a different scenario than the one in which Penrose makes his argument. We have seen that he has no problem in admitting the intellectual capabilities of machines. This issue is usually a common point among the detractors of the Penrosean perspective, and it seems to be a misunderstanding. What Penrose defends is that it is impossible for a machine to obtain a consciousness equal to that of a human being. That is the only capability that he denies to machines. Therefore, to attempt to imply that machines are denied any kind of intelligence is incorrect. Computers are very capable entities in numerous domains. It is another matter to try to equate such capabilities to that of humans. In some respects we humans are superior to machines and in others, the reverse is true. But the important thing for Penrose is not that, but to understand the gulf that irremediably separates us.

And the third point of Russell and Norvig's critique is basically the same as the first part of the second point. The question is raised again as to whether the human being is immune to the implications of the Gödelian theorem and how this does not delegitimize either of the two intelligences (Russell & Norvig, 2004: 1079). We have just seen how I disagree with this approach, so it would be a futile exercise to elaborate on my ideas in this regard.

The strength of the argument of Gödel's theorem can be affected by giving it a role that does not correspond to it. And it is precisely from this charge that Penrose does not escape. To argue that the Gödelian theorem has a factual utility as an argument against the computability of consciousness and the human mind is to dispense a burden to the theorem that is not really its own. But is this really so? It is true that it is essential to be cautious in establishing both epistemological and ontological equivalences between arguments from different fields. This, in my opinion, is not a problem for Penrose's approach. He makes it clear that formal and computational systems are deeply related, there being no such conflict (neither ontological nor epistemological).

Let us now turn to different critiques of Penrose's perspective in particular. We begin with those of Feferman and McDermott, which belong to a series of reviews by various authors, contained in the journal *Psyche* between 1995 and 1996. The reason for choosing these two criticisms in particular is because, I believe, they are the ones that contain a tone more antagonistic to what Penrose expressed.

Starting with Feferman, it should be noted that this author makes it clear from the beginning that the problem with Penrose's approach is not its content, but the way in which he intends to defend it. Moreover, Feferman even admits that he agrees with what Penrose defends.

One of the points that Feferman highlights is the clarity with which Penrose sees the relationship between formal systems and Turing machines. Although he shares the idea that the reformulation of the incompleteness theorem as an argument against understanding mathematical thought, on the other hand, he does not fail to question whether or not such an equivalence is forced. The reason for his doubt is that he thinks that Penrose gives formal systems a *modus operandi* that does not correspond to them and that clearly distances him from human beings. For Feferman it is not so clear that formal systems carry out mathematical thinking as described by Penrose. In fact, he argues that we cannot really guarantee knowledge of that *particular thing* that enables mathematical thinking (Feferman, 1995: 9).

Feferman shares with Penrose that *true* mathematical thinking is not mechanical and that "understanding" is definitive in this section, since it is in this notion that machines *could be* differentiated from human beings. However, the proof through Gödel's theorem does not have as much force as Penrose claims, but only translates a conviction that raises more questions than it answers (Feferman, 1995: 9).

Although Feferman offers a very concrete critique of Penrose's approach, he does not offer an alternative. What Feferman appeals to is Penrose's ambiguity in using certain concepts from the field of logic. As Penrose recognizes, Feferman is right in his criticism, but the solution is to understand such concepts in their most general way so as not to run an excessive risk. Penrose does not intend to enter into a purely logical debate. Let us now turn to McDermott's criticism.

McDermott's critique has a much less delicate tone than that professed by Feferman. The difference between the two attitudes probably lies in the fact that, unlike Feferman, McDermott does not share Penrose's point of view with respect to Artificial Intelligence. However, this is no reason for McDermott to criticize aspects closely related to those seen in the previous critique.

An example of this is to question Penrose's defense of the way mathematicians proceed, as far as thinking mathematics is concerned. For McDermott it is not so evident that there is an unbridgeable gulf between human beings and machines. In the first place, Penrose makes an imprudent mistake in generalizing about how mathematicians think; and, secondly, he makes the mistake of extrapolating Gödel's theorem to such an assumption.

The general tone of McDermott's critique is to critized Penrose for his lack of precision in terms that are fundamental for dealing with this type of debate (McDermott, 1995: 16).

McDermott sees many errors in Penrose's argument, but, in my opinion, we cannot see either any alternatives. On the one hand, although

Penrose gives a very important role to his Gödelian argument, he does not leave absolutely everything to it. And on the other hand, McDermott, after all, does not stop appealing to "in the future we will see what happens" (typical of those who defend AI), so that demanding more forcefulness from Penrose seems a demand, at least, out of place.

Penrose already in ENM realizes the possible criticisms that his position has to face. However, he considers that the arguments that may come his way are not strong enough, (Penrose, 1991: 517).

The reason he is not convinced by such arguments is that they do not accurately describe how human beings think about mathematics. For human beings, mathematics is not presented in such a way that it is impossible to give answers, no matter how abstract the problem is.

Despite the philosophical efforts of the authors we have seen, at present this type of criticism is not the main argument of those who defend Artificial Intelligence. Rather, they focus on expressing the enormous capabilities that the current machines have in order to, in this way, demolish the possible limitations that their detractors may want to grant them. A clear example of this trend can be found in Nick Bostrom (2014), who highlights the power of machines as follows:

> […] If the methods that the software uses to search for a solution are sufficiently sophisticated, they may include provisions for managing the search process itself in an intelligent manner […] the software may start by developing a plan for how to go about its search for a solution. The plan may specify which areas to explore first and with what methods, what data to gather, and how to make best use of available computational resources. In searching for a plan that satisfies the software's internal criterion […], the software may stumble on an unorthodox idea. For instance, it might generate a plan that begins with the acquisition of additional computational resources and the elimination of potential interrupters (such as human beings). Such "creative" plans come into view when the software's cognitive abilities reach a sufficiently high level. When the software puts such a plan into action, an existential catastrophe may ensue (Bostrom, 2014: 153).

Software sophistication has reached such a point that machine intelligence (superintelligence, as Bostrom calls it) in many ways resembles human intelligence (even today!). But to what extent can this way of thinking reflect such a degree of optimism without conflicting with

Penrose's perspective? Well, I think, as far as they want to. We have seen repeatedly that Penrose readily acknowledges the merits of technological advances. Therefore, a description such as Bostrom's does not argue against what Penrose advocates (i.e., that machines acquire human-like consciousness since it contains non-computational processes).

Now, what is Penrose's reason for adopting Gödel's theorem without that conviction being threatened by the criticism it may receive? In short, to mathematics. But to be more precise, to the relation of mathematics to truth. He understands that this field of knowledge allows us to have a direct contact with the truth. Therefore, this path must not be abandoned!

5. ACKNOWLEDGEMENTS

This work was funded by Next Generation EU, with a Margarita Salas contract (posdoctoral for young researchers).

6. REFERENCES

Bostrom, N. (2014). Superintelligence: Paths, Dangers, Strategies, Oxford, Oxford University Press.

Copeland, J. (2017). "Intelligent machinery", in J. Copeland, J. Bowen, M. Sprevak & R. Wilson, The Turing Guide, Oxford, Oxford University Press, pp. 265-276.

Detlefsen, M. (1996). "Philosophy of Mathematics in the twentieth century" in S. Shanker (ed.), Philosophy of Science, Logic and Mathematics in the twentieth century, Routledge, London, pp. 50-123.

Feferman, S. (1995). "Penrose's Gödelian Argument: A Review of Shadows of the Mind by Roger Penrose," Psyche, 2 (7), http://psyche.cs.monash.edu.au/v2/psyche-2-07-feferman.html.

Forouzan, B.A. (2003). Introducción a la ciencia de la computación: De la manipulación de datos a la teoría de la computación, trans. by Peralta, L., México D.F., International Thompson Editores, pp. 321-325.

Gödel, K. (2006). Obras completas, trans. by J. Mosterín, Madrid, Alianza Editorial, pp. 53-89.

Grattan-Guinness, I. (2017). "Turing's mentor, Max Newman", in J. Copeland, J. Bowen, M. Sprevak & R. Wilson, The Turing Guide, Oxford, Oxford University Press, pp.437-442.

Heredia, D. (2019). “La importancia del teorema gödeliano en el pensamiento de Roger Penrose”, en Naturaleza y Libertad (12), pp. 159-178.

Kennedy, J. (2017). "Turing, Gödel and the "Bright Abyss" in J. Floyd & A. Bokulich [eds.], Philosophical Explorations of the Legacy of Alan Turing: Turing 100, Boston, Springer, pp. 63- 92.

Ladrière, J. (1969). Limitaciones Internas de los Formalismos: Estudio sobre la significación del Teorema de Gödel y teoremas conexos en la teoría de los fundamentos de las matemáticas, trans. by J. Blaso, Madrid, Tecnos.

McDermott, D. (1995). "Penrose is wrong: A Review of Shadows of the mind by Roger Penrose", Psyche, 2(17), http://psyche.cs.monash.edu.au/v2/psyche2-17-mcdermott.html.

Penrose, R. (1991). La nueva mente del emperador, trans. by J. García, Barcelona, Grijalbo Mondadori.

(2012). Las sombras de la mente, trans. by J. García, Barcelona, Crítica.

Randell, B. (2017). "Turing and the origins of digital computers" in J. Copeland, J. Bowen, M. Sprevak & R. Wilson, The Turing Guide, Oxford, Oxford University Press, pp. 67-76.

Russell, B., Whitehead, A. N. (1997). Principia Mathematica, Cambridge, Cambridge University Press.

Russell, S. J., Norvig, P. (2004). Inteligencia Artificial: Un enfoque moderno, trans. by J. Rodríguez, F. Martín, J. Cadenas, L. Hernández, E. Paniagua, R. Fuentetaja, M. Robledo, and R. Rizo, Madrid, Pearson Educación S.A.