PH299

Essay on an Approved Subject in Philosophy

A Limit on Artificial Intelligence? – The Gödelian Case

Aarne Talman*

June 2005

Department of Philosophy, Logic and Scientific Method

The London School of Economics and Political Science

*I would like to thank Colin Howson, Wilfried Meyer-Viol and Panu Raatikainen for their advice and support.

Abstract

To show that there is a limitation on artificial intelligence it suffices to find evidence of intelligent behaviour in humans that is in principle impossible for any machine. It has been argued that Gödel's incompleteness theorems show that such a limitation exists. By Gödel's first incompleteness theorem there is for any consistent formal system satisfying some specific requirements a true sentence such that neither it nor its negation can be proved in that system. The claim is that we can somehow see that such Gödelian sentences are true and therefore the mental capacities of humans surpasses those of any machine. The aim of this essay is to review these Gödelian argumets against artificial intelligence and show that they are at most inconclusive.

1 Introduction

Artificial intelligence is the field of computer science that tries to model intelligent behaviour by computers. In particular the attempt is to build a computer that could perform tasks that are considered to be only within human mental capacity. Intelligence is, however, a tricky concept; it is very hard to draw a boundary between intelligent and unintelligent behaviour. Most people would, nevertheless, agree that us humans do possess intelligence. Hence to say something about the intelligence of a machine the most reasonable way is to compare its abilities to those of humans. In this sense artificial intelligence has been reasonably successful. Consider for example the chess-playing computer Deep Blue that beat the world chess champion Garry Casparov on May 1997. Clearly, to win a world champion in chess requires some sort of intelligence. However, no one would claim that the computers of today are as intelligent as humans. There are many things that we humans can do that require intelligence but which are unachievable for any existing computer. Hence a more important and more interesting question is whether it is in principle possible for a machine to be as intelligent as humans. How could this question be answered? One way to find a negative answer to the question is to look for evidence of intelligent behaviour in humans that is in principle impossible for a computer. It has indeed been argued that the incompleteness theorems proved by the great mathematical logician Kurt Gödel show that human mathematical thinking is unachievable for any computer.

By Gödel's first incompleteness theorem there is no complete consistent axiomatisation of mathematics or even of basic arithmetic. That is, for any consistent formal system satisfying some specific requirements there is a true sentence such that neither it nor its negation is provable from the set of axiom for that system. This theorem proved by Gödel in his famous *On formally undecidable propositions of Principia mathematica and related systems I* [GÖDEL 1931] has been argued by some people to prove that human mind is not a machine and that any attempt to model human mind by a computer is doomed to fail. Such arguments, the so-called Gödelian arguments, have been given most notably by Lucas [LUCAS 1961, LUCAS 1970, LUCAS 2000] and Penrose [PENROSE 1989, PENROSE 1994]. The aim of this essay is to review the Gödelian arguments against artificial intelligence.

In order to understand how the theorems of Gödel relate to computers and to answer the question whether the theorems place a limit on artificial intelligence I will first present a mathematical model of computers and computability known as Turing machines. I will then give an overview of the first and second incompleteness theorems and relate them to the mathematical model of computers. After the relation between computers and Gödel's theorems have been made clear I will move on to describe the Gödelian arguments by Lucas and Penrose. I will start with an argument that I call the Lucas-Penrose argument, and see how this arguments can be countered. After this I will present a new argument by Penrose that according to him is not subject to the same objections as the simple Lucas-Penrose argument. The new argument by Penrose can however be shown to be flawed; I give reasons for this claim at the end of section 5. Once it has been shown that neither the Lucas-Penrose argument nor the second argument by Penrose succeed, I will present Gödel's and Turing's ideas on the philosophical implications of Gödel's theorems. I will also present Gödel's own argument that the mental abilities of humans surpasses that of any idealised machine. The premises needed for Gödel's conclusion are, however, too philosophical and controversial in nature for them to be acceptable. Finally I will present an argument by Lucas that has been completely ignored in the literature. It is not directly based on Gödel's theorems but instead on a theorem by Tarski. I will conclude by showing that the argument does not succeed in establishing that there is a limitation on artificial intelligence.

2 Machines and Computability

As I mentioned above the presently existing computers cannot be considered to be as intelligent as humans. Therefore in order to say whether it is in principle possible or impossible for computers to be as intelligent as humans, it is crucial that we have a precise idealised model of computer at hand. In this section I give an overview of a mathematical model of computers and computability known as Turing machines[1] that was first developed by an English mathematician Alan Turing in his [TURING 1936]. My presentation follows [SIPSER 1997].

A Turing machine consists of a finite state control attached to read/write head on an infinite tape. The tape is divided into squares, each capable of storing one symbol at a time from a finite alphabet which includes the blank symbol. At each step in a computation the Turing machine is in one of the finite possible states. Initially the finite input string is written on adjacent squares of the tape. All the other squares on the tape are blank. The head scans the leftmost square of the input string, and the Turing machine is in a specific start state. At each step the machine is in some state and the head is scanning a tape square containing some symbol. The action performed depends on the symbol scanned by the machine and on the machine's state, and is specified by the machine's transition function. The action consists of printing a symbol on scanned square, moving the head left or right one square, and assuming a new state.

More formally, a Turing machine M is a 7-tuple $(Q, \Sigma, \Gamma, \delta, q_0, q_{accept}, q_{reject})$, where Q, Σ, Γ are finite sets:

1. Q is the set of states,
2. Σ is the input alphabet not containing the special *blank* symbol $\sqcup$,
3. Γ is the tape alphabet, where $\sqcup \in \Gamma$ and $\Sigma \subseteq \Gamma$,
4. $\delta : Q \times \Gamma \longrightarrow Q \times \Gamma \times \{L, R\}$ is the transition function (L and R abbreviate the read/write head's movement to the left and right respectively),

[1]There are many different mathematical models of computers and computability all of which have, however, turned out to be equivalent.

5. $q_0 \in Q$ is the start state,

6. $q_{accept} \in Q$ is the accept state,

7. $q_{reject} \in Q$ is the reject state, where $q_{reject} \neq q_{accept}$.

A configuration of M is a string xqy, where $x, y \in \Gamma^*$, y is not the empty string, $q \in Q$ and Γ^* is the set of all strings over Γ. When M has the configuration xqy, this means that M is in state q with xy on its tape and its head is scanning the leftmost symbol of y. Configuration C_1 is said to *yield* configuration C_2 if the Turing machine can legally go from C_1 to C_2. That is, the configuration vaq_ibu yields the configuration vq_jacu if in the transition function $\delta(q_i, b) = (q_j, c, L)$. In this case the Turing machine moves to the left. The configuration vq_ibu yields the configuration $vacq_ju$ if $\delta(q_i, b) = (q_j, c, R)$. In this case the Turing machine moves to the right. The configuration q of M is said to be *halting* if $q \in \{q_{accept}, q_{reject}\}$, the configuration is *accepting* if $q = q_{accept}$ and the configuration is *rejecting* if $q = q_{reject}$. A Turing machine M *accepts* input w if there is a sequence of configurations $C_1, C_2, \ldots, C_n$ such that: C_1 is the start configuration of M on input w, each C_i yields C_{i+1}, and C_n is an accepting configuration.

A Turing machine M computes as follows. First M receives its input $w = w_1 w_2 \ldots w_n \in \Sigma^*$ on the first n squares of the tape. The head starts on the leftmost square of the tape. The computation then proceeds according to the rules described by the transition function. The computation proceeds until M enters the halting state, otherwise M does not halt. It is usually convenient to take Σ to be some subset natural numbers.[2] Then a total n-place function $f : \mathbb{N}^n \to \mathbb{N}$ is said to be *recursive* if some Turing machine M, on every input $\vec{x} \in \mathbb{N}^n$, halts with just $f(\vec{x})$ on its tape. A relation P is recursive if there is a Turing machine M which given an input $\vec{x} \in \mathbb{N}^n$ after finite number of steps halts with number 1 on its tape if $P\vec{x}$ holds and number 0 on its tape if $\neg P\vec{x}$ holds. Similarly a set $S \subseteq \mathbb{N}$ is recursive if there is a Turing machine which given $\vec{x} \in \mathbb{N}^n$ as input after finite number of steps halts with number 1 on its tape if $\vec{x} \in S$ and number 0 on its tape if $\vec{x} \notin S$. A set $S \subseteq \mathbb{N}$ or a relation $P \subseteq \mathbb{N}^n$ is recursively enumerable iff it is in the domain of a recursive relation.

[2]Strictly speaking we should say that Σ is taken to be a subset of *numerals* for natural numbers, since Turing machines accept strings of symbols, not numbers.

Now that we have an understanding of Turing machines and how they operate, let us turn to the incompleteness theorems of Gödel and see how these theorems relate to Turing machines.

3 Gödel's Theorems

Gödel's theorems are about limitations of formal systems of arithmetic. A formal system is a set of axioms in some formal language together with an explicit definition of the notion of formal proof. The formal language of arithmetic L_A is usually taken to be the first-order language with equality such that it has the following non-logical symbols: $\mathbf{0}$ (for zero), $\boldsymbol{s}$ (for successor function), $+$ (for addition function) and $\times$ (for multiplication function). A formal system of arithmetic is a system that tries to formalise the theory $\mathrm{Th}(\mathfrak{N})$ of the natural number structure $\mathfrak{N} = (\mathbb{N}; 0, s, +, \times)$, that is the set of all sentences of L_A that are true in $\mathfrak{N}$.

In what follows we assume that the syntax of L_A has been coded in natural numbers. Such a coding is usually called Gödel numbering and the code number, $\#\varphi$, of a formula φ is called its Gödel number.

Now, the exact nature of formal proofs need not concern us here. What is, however, important is that proofs are finite, and if the set of Gödel numbers of the axioms of a formal system T is recursive, then the relation $T \vdash \varphi$, there is a proof of φ in T, coded in Gödel numbers is recursively enumerable. An important fact about such formal systems is that there is a Turing machine that given the set of Gödel numbers of axioms as input produces Gödel numbers of theorems of the system as outputs.[3] A formal system is said to be axiomatizable if it has a recursive set of axioms.

Let T be a formal system in the language L_A. Then there is a way in which we can express various facts about natural numbers in T. First of all we have a name for each natural number. The natural number m is denoted by $\boldsymbol{s}_m$. We can also express various relations on natural numbers by sentences of L_A. A formula ρ is said to *represent* a

[3]Henceforth instead of writing that a Turing machine, say M, produces as output the Gödel number of a sentence τ, I write that M proves τ. Also if a set of Gödel numbers of sentences is recursive or recursively enumerable I write that the set of sentences is recursive or recursively enumerable respectively.

relation R on natural numbers in T iff for every $a_1, \ldots, a_n$ in $\mathbb{N}$:

$$\text{if} \quad \langle a_1, \ldots, a_n \rangle \in R \quad \text{then} \quad T \vdash \rho(\boldsymbol{s}_{a_1}, \ldots, \boldsymbol{s}_{a_n}),$$

$$\text{if} \quad \langle a_1, \ldots, a_n \rangle \notin R \quad \text{then} \quad T \vdash \neg\rho(\boldsymbol{s}_{a_1}, \ldots, \boldsymbol{s}_{a_n}).$$

A relation is representable in T iff there is some formula that represents it there. A formula ρ is said to *weakly represent* a relation R on natural numbers in T iff for every $a_1, \ldots, a_n$ in $\mathbb{N}$:

$$\langle a_1, \ldots, a_n \rangle \in R \quad \text{iff} \quad T \vdash \rho(\boldsymbol{s}_{a_1}, \ldots, \boldsymbol{s}_{a_n}).$$

A relation is weakly representable in T iff there is some formula that weakly represents it there.

An interesting question from a logical point of view is whether there is a formal system T with a recursive set of axioms from which all and only the true sentences of arithmetic are provable. Such a formal system would be complete in the sense that for any sentence φ of L_A either $T \vdash \varphi$ or $T \vdash \neg\varphi$. By Gödel's first incompleteness theorem the answer is no if T is consistent, that is, if for no formula φ both $T \vdash \varphi$ and $T \vdash \neg\varphi$. First thing that has to be shown when proving the first incompleteness theorem for a formal system is that all recursive relations are representable in the formal system. If this was not the case then the formal system would already for that reason be incomplete. Such a formal system would also have to be sound in the sense that all sentences provable in it are true in $\mathfrak{N}$. Let us call a formal system sufficiently strong if all the recursive relations are representable in it. Gödel showed that a particular formal system known as Peano arithmetic, PA, is strong enough to represent all recursive relations. Peano arithmetic has six axioms and one axiom schema, these are given below:

1. $\forall x(\boldsymbol{s}x \neq 0)$
2. $\forall x \forall y(\boldsymbol{s}x = \boldsymbol{s}y \rightarrow x = y)$
3. $\forall x(x + 0 = x)$
4. $\forall x \forall y(x + \boldsymbol{s}y = \boldsymbol{s}(x + y))$
5. $\forall x(x \times 0 = 0)$

6. $\forall x \forall y(x \times \boldsymbol{s}y = x \times y + x)$

7. For each well-formed formula φ the following is an axiom

$$\varphi(0) \wedge \forall x(\varphi(x) \rightarrow \varphi(\boldsymbol{s}x)) \rightarrow \forall x \varphi(x)$$

Let us see how the first incompleteness theorem can be proved for this particular formal system. There are many ways to prove the first incompleteness theorem.[4] I will present the theorem for PA relating it explicitly to Turing machines. It is important to note that the theorems of PA are recursively enumerable. Hence we can construct a Turing machine M_{PA} that proves theorems of Peano arithmetic. We want to know whether the Turing machine M_{PA} proves all the true sentences of arithmetic. For this to be possible M_{PA} has to be *sound*, i.e. all sentences that it proves must be true, and complete in the sense that for all the sentences τ it proves either τ or $\neg\tau$.

Now, there is an important fact about formal systems known as the diagonal lemma or the fixed-point lemma. It states that if all the recursive relations are representable in a formal system T then given any formula ϕ with only one free variable we can find a sentence τ such that:

$$T \vdash (\tau \leftrightarrow \varphi(\boldsymbol{s}_{\#\tau})).$$

Once this lemma has been proved the first incompleteness theorem can be proved easily. Consider the relation $\mathrm{PA} \vdash \varphi$. If this relation holds, then M_{PA} proves φ. The relation is weakly representable in PA, so let $\exists x \varrho(\boldsymbol{s}_{\#\varphi}, x)$ weakly represent it there.[5] Let $\mathrm{Prb}_{\mathrm{PA}}\varphi = \exists x \varrho(\boldsymbol{s}_{\#\varphi}, x)$. Now, we are interested in the claims of the form $\mathrm{PA} \nvdash \varphi$, i.e. M_{PA} does not prove φ. By the fixed-point lemma there exists a sentence σ of L_A such that $\mathrm{PA} \vdash (\sigma \leftrightarrow \neg\mathrm{Prb}_{\mathrm{PA}}\sigma)$. Now we ask the question: does M_{PA} prove σ? That is, does the relation $\mathrm{PA} \vdash \sigma$ hold? Assume that it does, then clearly $\mathrm{PA} \vdash \mathrm{Prb}_{\mathrm{PA}}\sigma$. From the instance of the fixed-point lemma we also obtain $\mathrm{PA} \vdash \neg\mathrm{Prb}_{\mathrm{PA}}\sigma$. But if this is the case it would immediately follow that M_{PA} is not sound. So $\mathrm{PA} \vdash \neg\sigma$ must hold. But if $\mathrm{PA} \vdash \neg\sigma$ holds then $\mathrm{PA} \vdash \neg\mathrm{Prb}_{\mathrm{PA}}\sigma$ also holds. From the fixed-point lemma one obtains also that $\mathrm{PA} \vdash \mathrm{Prb}_{\mathrm{PA}}\sigma$, which again contradicts the requirement of soundness. So it follows that

[4] For a nice discussion of the different ways to prove the theorem see [ENDERTON 2001].

[5] For the details see [ENDERTON 2001].

there is a sentence σ such that neither σ nor $\neg\sigma$ is provable from PA. Hence neither of them can be proved by a sound Turing machine M_{PA}.

But why do we require M_{PA} to be sound? What if M_{PA} is not sound? Then we would not take M_{PA} to be adequate for proving theorems of arithmetic, for clearly we want those theorems to be true. Note that the first incompleteness theorem holds even if we change the requirement of soundness to a weaker requirement of consistency. A sentence σ such that $\mathrm{PA} \vdash (\sigma \leftrightarrow \neg \mathrm{Prb}_{\mathrm{PA}}\sigma)$ is called a Gödel sentence for PA. Note also that if we add the sentence σ as an axiom of PA then there is a new Gödel sentence for $\mathrm{PA} \cup \{\sigma\}$, say σ', such that neither $\mathrm{PA} \cup \{\sigma\} \vdash \sigma'$ nor $\mathrm{PA} \cup \{\sigma\} \vdash \neg\sigma'$.

There is one important fact that needs to be made clear at this point. The first incompleteness theorem holds for any sufficiently strong axiomatizable consistent formal system in any formal language. Hence we cannot make PA complete by adding new axioms in it. It is also not just PA and theories containing PA that are subject to the theorem; any formal system that satisfies the requirements mentioned above is incomplete.[6] The first incompleteness theorem places obviously a limitation on any Turing machine that is "powerful" enough to prove theorems of a sufficiently strong formal system. For any such Turing machine there is a sentence that the Turing machine cannot prove or disprove.

Before we go into the Gödelian arguments let us look at the second incompleteness theorem. The theorem will be used when discussing the Gödelian arguments. Gödel's second incompleteness theorem says that the consistency or inconsistency of a consistent formal system extending PA cannot be proved in the system itself.[7] To put the same more formally, assume that T is a consistent formal system extending PA. Assume also that T axiomatizable. Then $T \nvdash \mathrm{Cons}(T)$ and $T \nvdash \neg\mathrm{Cons}(T)$, where $\mathrm{Cons}(T)$ is a formula of L_A expressing the consistency of T in T, say the formula $\neg\mathrm{Prb}_T(\mathbf{0} = \boldsymbol{s}\mathbf{0})$. Notice that clearly an inconsistent formal system proves its own consistency, for it proves every sentence.

---

[6] In fact there are formal systems of arithmetic that are weaker than PA but which satisfy these requirements. One such system is called Robinson arithmetic, Q. It can be obtained from PA by replacing axiom schema 7 by the sentence $\forall x(x \neq \mathbf{0} \rightarrow \exists y(y = \boldsymbol{s}y))$.

[7] A formal system T is said to be an extension of PA iff everything that can be proved in PA can also be proved in T.

4 The Lucas-Penrose Argument

The basic idea behind all the Gödelian arguments is the claim that by the first incompleteness theorem we can find for any consistent sufficiently strong formal system that has a recursive set of axioms a true sentence such that both it and its negation is unprovable in the system, but such that we humans can recognise as true. If this was the case then, it is argued, it would immediately follow that human mathematical intelligence cannot be modelled by any machine and hence there is a limit to what can be achieved by artificial intelligence.

The first explicit formulation of a Gödelian argument can be found in a much quoted article by John Lucas [LUCAS 1961].[8] One can, however, find remarks about the possibility to use such an argument already in Turing's ground-breaking article on artificial intelligence [TURING 1950] and in a popular book on Gödel's theorems by Nagel and Newman [NAGEL AND NEWMAN 1958]. More recently Roger Penrose has defended a similar Gödelian argument in his [PENROSE 1989] and [PENROSE 1994], hence I will call the argument the Lucas-Penrose argument.

The Lucas-Penrose argument is simple. One of the formulations of it by Lucas goes as follows:

> [w]e now construct a [Gödel sentence] in [a formal system], say L, which cannot itself be [proved in L]. Therefore the particular human being who is [...] represented by [L] cannot produce such a formula as being true. But he *can* see that it is true: any rational being could follow Gödel's argument, and convince himself that the [Gödel sentence], although [unprovable in L] was nevertheless – in fact, for that very reason – true. Therefore human being cannot be represented by a [formal system] [LUCAS 1970, p. 133, original emphasis].

So given any machine that proves theorems of a formal system there is a Gödel sentence for that system that cannot be proved by the machine, but which we can somehow see to be true. Essentially the same argument was presented much later by Penrose. According to Penrose we can ascertain the truth of a Gödel sentence by conscious contemplation

[8] See also [LUCAS 1970] and [LUCAS 2000].

and since we can always find a Gödel sentence for any formal system, our mathematical reasoning and insight must be non-algorithmic, i.e. impossible to be modelled by a Turing machine [PENROSE 1989].

It is probably best to understand the argument in the following formulation. Assume that a sound Turing machine M_i proves all the mathematical theorems that us humans can in principle know to be true. Then by the Gödel's theorem human mathematicians can find a sentence that M_i cannot prove but such that humans can see to be true. But this contradicts the claim that M_i can prove everything humans can see to be true. And since we made no assumptions about the nature of M_i (other than soundness) there cannot be any such Turing machine. So artificial intelligence has a limitation.

Notice that the claim by Lucas and Penrose is a very strong one and if true, it is of great philosophical importance. Not only does it show that artificial intelligence is limited, it also shows something about the nature of human mind, i.e. that it is not a machine.

Let us now see how the Lucas Penrose argument can be countered. One of the earliest objetions to the Lucas-Penrose argument was given by Hilary Putnam. Putnam formulated the Gödelian argument independently of Lucas but his objection has become a standard objection against the Lucas-Penrose argument. To understand Putnam's objection let us first look at the Gödel sentence for PA in more detail. Is σ produced in the previous section true? Firstly, notice that PA has to be at least consistent for σ to be unprovable. Secondly, σ is provably equivalent to the claim in PA that no number is the Gödel number of the proof of σ in PA. So if PA is consistent then σ is unprovable and hence, being equivalent to a sentence indirectly stating its unprovability in PA, it must be true. But now notice that by the second incompleteness theorem the consistency of PA cannot be proved in PA. So we do not know whether σ is true or not. Moreover, by the second incompleteness theorem it follows that $\mathrm{PA} \vdash (\mathrm{Cons}(\mathrm{PA}) \rightarrow \sigma)$. So M_{PA} proves $\mathrm{Cons}(\mathrm{PA}) \rightarrow \sigma$. What Putnam pointed out was that a Gödel sentence σ of a formal system, say T, that cannot be proved by a Turing machine cannot be proved by humans either. All we can prove is $\mathrm{Cons}(T) \rightarrow \sigma$, but as much can already be proved by a Turing machine. Hence the Lucas-Penrose argument fails.

There are a few ways one can go about trying to "prove" consistency of a formal system such as PA.

First, one could argue that people have used principles that are formalised in PA for centuries, in fact most of the mathematics used in our everyday lives can be proved in PA, and so far no inconsistencies have occurred. Hence it is reasonable to believe that PA is consistent. In other words we have strong *inductive* support for the claim that PA is consistent. The same line of reasoning could be applied to other well known formal systems, for example ZF or ZFC set theory. No inconsistencies have been found in these formal systems, even though mathematicians have explicitly been trying to find one.

Such an inductive argument for the consistency of a formal system is, however, inconclusive. No matter how much "evidence" we have for the consistency of a formal system it can still turn out to be inconsistent. The history of mathematics and logic has its examples of formal systems that seemed to be consistent but in the end turned out not to be. Frege's system is one example.[9] Now, remember that the claim was that Gödel's theorems show, or even prove, that artificial intelligence has a limitation. But if we use an inductive argument to do the work in the proof, then we might as well forget about the Gödel's theorems and argue that we have strong inductive support that artificial intelligence is limited, for no one has so far been able to build a machine that is as intelligent as humans. But clearly we would not accept this as a plausible argument against artificial intelligence. Hence one cannot really accept inductive support to back up the Lucas-Penrose argument.

Second, one could argue that clearly PA is consistent by using a so-called semantical argument. All one needs to do is to notice the following fact of logic known as the soundness of predicate calculus: if a set of sentences has a model then it is consistent. That is, if all the members of a set of sentences are true in a structure, then the set is consistent. But clearly the axioms of PA are true in the natural number structure $\mathfrak{N}$, so clearly PA is consistent and thus the Gödel sentence for PA is true.

But this is too fast. By the soundness of predicate calculus together with the completeness theorem, which was also proved by Gödel in his doctoral dissertation [GÖDEL 1929], it follows that a formal system is consistent if and only if the system has a model. But now since we cannot prove the consistency of any consistent sufficiently strong formal

[9] See e.g. [FREGE 1879].

system, it follows that we cannot prove that a model for a formal system exists. Hence the claim that PA is consistent because it has a model is question-begging.

One can of course prove the consistency of PA in some stronger formal system such as ZFC set theory (on relative consistency proofs see e.g. [KUNEN 1980]). So a Turing machine that proves theorems of ZFC, call it M_{ZFC}, proves a sentence equivalent to $\mathrm{Cons(PA)}$. So M_{ZFC} proves a Gödel sentence for PA. But now what if ZFC is itself inconsistent? Then it does prove some false sentences and hence the Turing machine proving theorems of ZFC is unsound. Again by the second incompleteness theorem the consistency of ZFC cannot be proved in ZFC unless ZFC is inconsistent. This time it is not really possible to use a semantical argument to prove the consistency of ZFC, since it is not really clear whether there is a model of ZFC or not. Moreover, as has been argued by Boolos, the Lucas-Penrose argument fails since the claim that we can see the Gödel sentence for *any* given formal system to be true is unwarranted [BOOLOS 1990]. It is totally different issue to claim that a relatively simple formal system such as PA can be seen to be consistent and hence its Gödel sentence to be true than to claim that we can somehow ascertain that some more complicated formal systems, such as ZFC, is consistent and hence its Gödel sentence is true.

There is a way that one can argue against the Lucas-Penrose argument even without mentioning consistency. That is, the Lucas-Penrose argument can be countered even if we grant that we can somehow see that a Gödel sentence for a formal system is true. This can be done by turning the Lucas-Penrose argument upside-down. Remember that the claim was that given any Turing machine that proves theorems of a certain formal system we can always find a true sentence that cannot be proved by the machine. But notice that we could also argue that every time we use a Gödelian argument, a new machine can be constructed that proves exactly the theorems that we can see to be true. Since this process could in principle be continued *ad infinitum*, at no stage can we say that there is *no* machine that proves exactly the sentences we can see to be true. We can only claim that some particular machine is not sufficient for proving all the sentences we can see to be true. From this it follows that given any Gödel sentence there is a Turing machine that proves it. Hence the Lucas-Penrose argument fails. This objection was given already by

Turing in his [TURING 1950].[10]

Lucas has offered an argument[11] against the last objection in his [LUCAS 1970]. His claim is that it must be in principle possible to specify a single machine that can prove everything we can prove, and once we specify the machine we can apply the Lucas-Penrose argument to show that there is a sentence that we can see to be true but such that the machine cannot prove. He writes: '[t]he determinist cannot say "that a machine could do that as well" for it is *another* machine, and what we were arguing about was whether the machine or physical system *as originally specified* was equivalent to a man' [LUCAS 1970, p. 142, original emphasis]. Lucas's reasoning misses, however, the point. The issue is not whether it is some particular machine that can prove everything we can, but whether there *is* a machine that can do this, even if we cannot specify one. This has not been shown to be impossible by the Lucas-Penrose argument.

As we can now see the Lucas-Penrose argument fails mainly because it rests on the unwarranted claim that we can somehow see the truth of Gödel sentences, but it fails also beacause even if we could see the truth of a Gödel sentence there is always a possibility to construct a new machine that can prove that Gödel sentence too.

5 Penrose's Second Argument

Penrose has presented an another Gödelian argument in his [PENROSE 1994] that according to him does not depend on our capabilities of ascertaining the consistency or soundness of any formal system. I will present the argument in more condensed form than Penrose himself does. My presentation is based on [LINDSTRÖM 2001] and [SHAPIRO 2003].

The argument can be formulated as follows. Assume that some Turing machine M_i is able to prove all the mathematical theorems I can prove. Let us abbreviate this claim by $I = M_i$. If $I = M_i$ then it follows that M_i is sound, since I know that I am sound. But if I know that $I = M_i$, then M_i^+ is sound, where M_i^+ is M_i together with the claim that M_i is sound. Now I know that if M_i^+ is sound then the Gödel sentence for the formal system

[10] Cf. section 6.

[11] His argument is targeted against determinism, but it applies to the present discussion as well.

whose theorems M_i^+ proves is true. But I also know that M_i^+ cannot prove this Gödel sentence. Now since we assumed that $I = M_i$ it follows that $I = M_i^+$, for $I = M_i$ and I know that $I = M_i$. Also I know that a Gödel sentence for M_i^+ is true. But this cannot be if $I = M_i^+$. So from the assumptions that $I = M_i$ it follows both $I = M_i^+$ and $I \neq M_i^+$, which is a contradiction. So the assumption $I = M_i$ must be false. So there cannot be a sound Turing machine that proves all the mathematical theorems I can prove.

There are couple of ways to counter this argument.

First, notice that the argument rests on the claim that from $I = M_i$ it follows that I can see that M_i^+ is sound. But this is not the case. Penrose assumes "I know that I am sound" which is, to say the least, dubious. Also, it does not follow from $I = M_i$ that I know that $I = M_i$. Hence it does not follow from $I = M_i$ and "I know that I am sound" that I know that M_i^+ is sound, for this assumes that I know that $I = M_i$. So it does not follow that I know that a Gödel sentence for M_i^+ is true. Therefore it can be concluded that Penrose's *reductio ad absurdum* fails.

A second way to counter Penrose's second argument is to argue similarly as in the end of the previous section. That is, the issue is not whether it is some particular machine that can prove everything I can, but whether there *is* a machine that can do this, even if we cannot specify one. As we saw in the previous section for any Gödel sentence there is a Turing machine that proves it.

6 Turing and Gödel on Gödelian Arguments

As I mentioned at the beginning of section 4, Turing had already noticed the possibility to argue against artificial intelligence using a Gödelian argument in his [TURING 1950]. Turing however abandoned the argument as flawed. He writes:

> [t]he short answer to this argument is that 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 [TURING 1950, p. 451].

Turing also writes that

> our superiority can only be felt on such an occasion in relation to the one machine over which we have scored our petty triumph. There would be no question of triumphing simultaneously over *all* machines. In short, then, there might be men cleverer than any given machine, but then again there might be other machines cleverer again, and so on [TURING 1950, p. 451, original emphasis].

This is exactly the same objection as we saw in the end of section 4 above.

Gödel made a similar kind of observation as Turing in his [GÖDEL 1951]. According to Gödel, his incompleteness theorems do not by themselves imply that artificial intelligence has a limitation. He noted that all we can derive from his theorems is that either '*the human mind infinitely surpasses the powers of any finite machine*' and hence there is a limitation to what can be achieved by artificial intelligence, '*or else there exist absolutely unsolvable diophantine problems*'[12] [GÖDEL 1951, p. 310, original emphasis]. Gödel thought that some additional premise is needed in order to decide between these two possibilities. Gödel suggested three such premises: (1) what he called "rationalistic optimism", (2) that mind is not static but constantly developing, and (3) that mind is separate from matter [WANG 1995]. Gödel's own conviction was that there cannot be humanly unsolvable diophantine problems, for he thought that human mind would be irrational if it would ask questions it could not, even in principle, find an answer. This is what he called "rationalistic optimism". He also believed that in the future (3) would be scientifically proved [WANG 1995].

As we can see Gödel's own argument rests on highly philosophical premises and hence does not have the power as the Gödelian arguments discussed above. But because of its philosophical nature it is much harder to refute than the arguments by Lucas and Penrose.

The claim (2) is probably the most easy to refute, for even if mind was constantly developing it does not mean that a machine cannot be. In fact there is a field of artificial intelligence called "machine learning" that tries to develop ways in which machine could

[12] A diophantine problem is to find a solution to $p(x_1, \ldots, x_n) = 0$, for a polynomial p with integer coefficients. By Gödel's incompleteness theorems and what is called MDRP theorem it follows that there are diophantine problems which are undecidable by any Turing machine. See [BELL AND MACHOVER 1977] for details.

be programmed to learn.

From claim (3) if true it only follows that materialism is false, but it does not place a limitation on artificial intelligence. For clearly even if mind was distinct from matter it does not follow that it surpasses any machine. Hence for the present discussion (3) is irrelevant.

Let us turn to claim (1). Gödel claims that even if no Turing machine is able to solve all diophantine problems, humans are, at least in principle, able to do so. It is easy to see that this premise, if true, guarantees that 'the human mind surpasses the powers of any finite machine'. In fact, both the premise and the conclusion seem to claim almost the same thing. Hence, without a good argument why we should accept the premise as true, the argument establishes nothing. But the only argument Gödel seems to give for his premise is the claim that if it is not true human mind would be irrational in asking questions it cannot in principle find an answer. But clearly this is not enough to establish claim (1).

It can, therefore, be concluded that the argument by Gödel cannot be accepted without a plausible argument for rationalistic optimism and for the same reason the argument is at most inconclusive.

7 An Argument from Tarski's Theorem

Lucas has presented in his [LUCAS 2000] an interesting new argument against mechanism that does not explicitly use Gödel's theorems and that has been completely ignored in the literature. Instead of Gödel's theorems the new argument is based on Tarski's theorem. By Tarski's theorem the truth predicate is not representable in $\mathrm{Th}(\mathfrak{N})$.[13] Tarski's theorem can be seen as a corollary to Gödel's first incompleteness theorem or alternatively one can prove it directly from the fixed-point lemma. To see how this can be done, assume for a contradiction that the truth predicate, i.e. the predicate Tr such that $Tr(\phi)$ holds iff $\phi \in \mathrm{Th}(\mathfrak{N})$, is representable in $\mathrm{Th}(\mathfrak{N})$. Let τ represent Tr in $\mathrm{Th}(\mathfrak{N})$. Then by

[13]Note that the notation $\mathrm{Th}(\mathfrak{N})$ is language dependent, it is the set of sentences of L_A that are true in $\mathfrak{N}$. Hence Tarski's theorem shows that the truth predicate is not representable in the theory of natural number structure in the language of arithmetic L_A.

the fixed-point lemma there is a sentence ϱ such that $\varrho \in \mathrm{Th}(\mathfrak{N})$ iff $\neg\tau(\boldsymbol{s}_{\#\varrho}) \in \mathrm{Th}(\mathfrak{N})$. But the truth predicate also allows us to conclude that $\varrho \in \mathrm{Th}(\mathfrak{N})$ iff $\tau(\boldsymbol{s}_{\#\varrho}) \in \mathrm{Th}(\mathfrak{N})$. From which we can conclude that $\tau(\boldsymbol{s}_{\#\varrho}) \in \mathrm{Th}(\mathfrak{N})$ iff $\neg\tau(\boldsymbol{s}_{\#\varrho}) \in \mathrm{Th}(\mathfrak{N})$, from which a contradiction follows. Hence it can be concluded that the truth predicate is not representable in $\mathrm{Th}(\mathfrak{N})$, and hence not in any consistent sufficiently strong formal system in the language L_A either. The argument above is a formalisation of the well-known liar paradox, which arises when one asks: is the sentence 'this sentence is false' true?

Lucas's argument using Tarski's theorem is the following: 'since truth cannot be adequately expressed in a formal system, and since we evidently possess the concept of truth, we cannot be just the embodiment of some formal system' [LUCAS 2000, p. 218]. Lucas continues that '[t]he mechanist who wishes to deny this, would be reduced to denying that we really have a concept of truth at all. Some are prepared to maintain just this, but then have difficulty in claiming that mechanism is true' [LUCAS 2000, p. 218].

If valid this argument clearly places a limitation on artificial intelligence, for "possessing the concept of truth" can be regarded as a sign of intelligence.

But what does possessing the concept of truth mean? There are at least two ways we can understand Lucas's claim. First, we have the concept of truth and given any sentence α in some language L, the concept of truth applies either to α or the negation of α. This reading is, however, untenable because of the liar paradox. As we saw above there are sentences in natural languages, say in English, such that the concept of truth in that language does not apply to the sentence nor its negation. Hence we cannot possess the "full" concept of truth. Second, we can understand possessing the concept of truth as possessing a concept that is like the concept of truth but such that it fails to apply in some cases. We evidently *do* possess such a concept, for there are unproblematic cases where we can use the concept of truth. For example "all bachelors are unmarried" is a true sentence in English.

Now, in spite of Tarski's theorem there is a way in which truth can be expressed in a formal system. To see this the class of all formulas of L_A need to be divided into subclasses. The classes Σ_n^0, Π_n^0 and Δ_n^0 of relations on natural numbers are first defined as follows: Σ_0^0 is the class of all recursive relations. For all n, Π_n^0 consists of all relations which are negations of relations belonging to Σ_n^0. For all n, $\Delta_n^0 = \Sigma_n^0 \cap \Pi_n^0$. For all n, Σ_{n+1}^0

consists of all relations obtained by existential quantification from relations belonging to Π^0_n. A formula that represents in $\mathrm{Th}(\mathfrak{N})$ a relation in Σ^0_n or Π^0_n is called a Σ^0_n-formula or Π^0_n-formula respectively. Similarly for Δ^0_n. Given this classification of formulas of L_A and a consistent sufficiently strong formal system T there is the following result. For each n there is a formula $Tr_{\Sigma^0_n}$ such that for all Σ^0_n-sentences φ

$$T \vdash (\varphi \leftrightarrow Tr_{\Sigma^0_n}(\boldsymbol{s}_{\#\varphi})).$$

Similarly, for each n there is a formula $Tr_{\Pi^0_n}$ such that for all Π^0_n-sentences φ

$$T \vdash (\varphi \leftrightarrow Tr_{\Pi^0_n}(\boldsymbol{s}_{\#\varphi})).$$

For each n the formulas $Tr_{\Sigma^0_n}$ and $Tr_{\Pi^0_n}$ are called partial truth predicates for Σ^0_n-sentences and Π^0_n-sentences respectively (see [LINDSTRÖM 2003] or [SMORÝNSKI 1977] for more details). This result shows that partial truth predicates for sentences of L_A *can* be expressed in a formal system of arithmetic.

As we saw Tarski's theorem implies that no consistent formal system in L_A can possess the "full" concept of truth in L_A. By the liar paradox a similar result holds for humans as well. Hence the truth predicate we possess must be only partial, but as we saw formal systems can also possess partial truth predicates. So even if we do not possess the full truth predicate there is no difficulty in claiming that something is true or false. Therefore, it must be concluded that the argument by Lucas fails to show that there is a distinction between human mind and formal systems and hence that there is a limitation on artificial intelligence.

8 Conclusions

We have seen above four arguments for the claim that the theorems of Gödel and Tarski place a limit on artificial intelligence. The first argument was the Lucas-Penrose argument, which fails firstly because the claim that we can see a Gödel sentence σ for a formal system T to be true does not follow from Gödel's theorems. All we can conclude from the theorems is $\mathrm{Cons}(T) \rightarrow \sigma$ which can already proved in T. Furthermore, we saw that even if there is a way in which we can see σ to be true there ia always a Turing machine

that can also "see" that σ is true (by proving it). The second argument by Penrose also fails, for it rests on the false assumption that from $I = M_i$ it follows that I can see that M_i^+ is sound, which we saw not to be the case. The argument by Gödel was shown to be inconclusive for the reason that it rests on Gödel's rationalistic optimism, which has not been shown to be true. Finally the argument by Lucas using Tarski's theorem was shown to be flawed for the reason that, unlike what Lucas seems to assume, we do not possess a "full" truth predicate that would apply to any sentence or its negation, the contrary follows from the liar paradox. From this it follows that we must possess only a partial truth predicate. But as we saw, a formal systems can also possess such partial truth predicates.

All this does not, of course, mean that artificial intelligence has no limitations and hence that computers can in principle be as intelligent as humans. All that has been shown is that the arguments using Gödel's incompleteness theorems or Tarski's theorem fail to show that there is such a limitation.

References

[BARWISE 1977] Barwise, J. (ed.)(1977), *Handbook of Mathematical Logic*, Amsterdam: North-Holland.

[BELL AND MACHOVER 1977] Bell, J. L. and Machover, M. (1977), *A Course in Mathematical Logic*, Amsterdam: North-Holland.

[BOOLOS 1990] Boolos, G. (1990), 'On "Seeing" the Truth of the Gödel Sentence', reprinted in G. Boolos 1998.

[BOOLOS 1998] Boolos, G. (1998), *Logic, Logic and Logic*, Cambridge, Massachusetts: Harvard University Press.

[COPELAND 2004] Copeland, B. J. (ed.)(2004), *The Essential Turing*, Oxford: Oxford University Press.

[ENDERTON 2001] Enderton, H. B. (2001), *A Mathematical Introduction to Logic*, 2nd edn. New York: Academic Press.

[FREGE 1879] Frege, G. (1879), 'Begriffsschrift', reprinted in J. van Heijenoort 1967.

[GÖDEL 1929] Gödel, K. (1929), 'On the completeness of the calculus of logic', reprinted in K. Gödel 1986.

[GÖDEL 1931] Gödel, K. (1931), 'On formally undecidable propositions of *Principia mathematica* and related systems I', reprinted in K. Gödel 1986.

[GÖDEL 1951] Gödel, K. (1951), 'Some basic theorems on the foundations of mathematics and their implications', reprinted in K. Gödel 1995.

[GÖDEL 1986] Gödel, K. (1986), *Collected Works*, Volume I, Oxford: Oxford University Press.

[GÖDEL 1995] Gödel, K. (1995), *Collected Works*, Volume III, Oxford: Oxford University Press.

[KUNEN 1980] Kunen, K. (1980), *Set Theory. An Introduction to Independence Proofs*, Amsterdam: Elsevier.

[LINDSTRÖM 2001] Lindström, P. (2001), 'Penrose's new argument', *Journal of Philosophical Logic* **30**, pp. 241-250.

[LINDSTRÖM 2003] Lindström, P. (2003), *Aspects of Incompleteness*, 2nd edn. Natic, Massachusetts: Association for Symbolic Logic and A K Peters.

[LUCAS 1961] Lucas, J. R. (1961), 'Minds, Machines and Gödel', *Philosophy* **36**, pp. 112-137.

[LUCAS 1970] Lucas, J. R. (1970), *The Freedom of the Will*, Oxford: Oxford University Press.

[LUCAS 2000] Lucas, J. R. (2000), *The Conceptual Roots of Mathematics*, London: Routledge.

[NAGEL AND NEWMAN 1958] Nagel, E. and Newman, J. R. (1958), *Gödel's Proof*, New York: New York Universty Press.

[PENROSE 1989] Penrose, R. (1989), *The Emperor's New Mind*, Oxford: Oxford University Press.

[PENROSE 1994] Penrose, R. (1994), *Shadows of the Mind*, London: Vintage.

[PUTNAM 1960] Putnam, H. (1960), 'Minds and Machines', reprinted in H. Putnam 1975.

[PUTNAM 1975] Putnam, H. (1975), *Mind, Language and Reality*, Cambridge: Cambridge University Press.

[SHAPIRO 2003] Shapiro, S. (2003), 'Mechanism, Truth, and Penrose's New Argument', *Journal of Philosophical Logic* **32**, pp. 19-42.

[SIPSER 1997] Sipser, M. (1997), *Intoduction to the Theory of Computation*, Boston: PWS Publishing Company.

[SMORÝNSKI 1977] Smorýnski, C. (1977), 'The Incompleteness Theorems', in J. Barwise 1977.

[TARSKI 1935] Tarski, A. (1935), 'The Concept of Truth in Formalized Languages', in A. Tarski 1983.

[TARSKI 1983] Tarski, A. (1983), *Logic, Semantics, Metamathematics*, 2nd edition, edited by J. Corcoran, Indianapolis, Indiana: Hackett Publishing Company.

[TURING 1936] Turing, A. (1936), 'On computable numbers, with an application to the Entscheidungsproblem', reprinted in B. J. Copeland 2004.

[TURING 1950] Turing, A. (1950), 'Computing Machinery and Intelligence', reprinted in B. J. Copeland 2004.

[VAN HEIJENOORT 1967] van Heijenoort, J. (ed.)(1967), *From Frege to Gödel. A Source Book in Mathematical Logic, 1879-1931*, Cambridge, Massachusetts: Harvard University Press.

[WANG 1974] Wang, H. (1974), *From Mathematics to Philosophy*, London: Routledge and Kegan Paul.

[WANG 1995] Wang, H. (1995), *Reflections on Kurt Gödel*, Cambridge, Massachusetts: MIT Press.