LUNDS UNIVERSITET

Lucas and Penrose vs. Computationalism:
A Refutation of the Gödel-arguments

By Leonard Nygren Löhndorf, Spring 2021

Lund University
Department of Philosophy
Bachelor's Thesis

Date of the defense: June 24, 2021
Thesis advisor: Robin Stenwall
FTEK01

Abstract

John Lucas and Roger Penrose attempted to refute Computationalism, the theory that the human mind is a computational system. They did so by utilizing meta-mathematical proofs by Kurt Gödel. I will explore various objections made against the attempt and discover that it relies on two uncertain premises – human consistency and proof for the consistency of a formal system. Revisions of the argument are made to avoid the second premise, but the first premise remains. In addition to the assumed premises, various technical objections have been made against Lucas' and Penrose's arguments. Lucas and Penrose's attempt at refuting Computationalism ultimately fails because of the technical objections and the lack of a cogent proof for human consistency.

Contents

Abstract **2**

1. Introduction **4**

2. Theory **5**

2.1 Formal languages, formal systems, and interpretations 5

2.1.1 Peano arithmetic 6

2.2 Turing Machines and the Church-Turing thesis 6

2.3 Decidability and arithmetization 7

2.4 Gödel-1 8

2.5 The halting problem 8

2.6 Gödel-2 10

3. The Lucas-Penrose argument **10**

3.1 The Lucas-Penrose argument 10

3.1.1 The Lucas-Penrose argument formalized 11

3.2 Minds, Machines and Gödel – John Lucas 11

3.3 Objections by Paul Benacerraf 13

3.3.1 Lucas's rejoinder to Benacerraf 14

3.3.2 Lucas and Benacerraf: discussion 15

3.4 Objections by David Lewis 16

3.4.1 Lucas' rejoinder to Lewis 17

3.4.2 Lucas and Lewis: discussion 17

3.5 Concluding comments on the Lucas-Penrose argument 18

3.5.1 Reiteration of the Lucas-Penrose argument and its implications 18

3.5.2 The Lucas-Penrose argument: conclusion 18

4. The new Penrose argument **19**

4.1 The Penrose-argument 19

4.1.1 What the Penrose-argument escapes 20

4.1.2 The Penrose-argument formalized 20

4.2 Chalmers' objection 21

4.2.1 Penrose's response 22

4.2.2 Lindström's response 22

4.2.3 Penrose, Chalmers and Lindström: discussion 22

4.3 Lindström's objection 23

4.4 Bringsjord and Xiao's objection 23

4.5 The Penrose-argument: conclusion 24

5. Summary and Conclusion **24**

References **25**

1. Introduction

The *Computational Theory of Mind*[1] (CTM) is the theory stating that the human mind is a system of computations (Rescorla 2020). If the theory proves to be true, it entails that human consciousness exists due to some form of computation, and furthermore, that the human mind and its consciousness can in principle be emulated by a computational device, i.e., a machine. The subject raises questions such as – "is the human mind 'merely' a strong machine?".

Different arguments have been made against the CTM. Some of which have been based on Gödel's two incompleteness theorems ("Gödel-1" and "Gödel-2")[2]. Arguments against CTM that are based on Gödel's results will be called "Gödel-arguments", and it is the Gödel-arguments by John Lucas (1961) and Roger Penrose (1989; and 1994) this paper will be focused on. In this paper it is argued that the Gödel-arguments fail, as they are flawed. The Gödel-arguments require the assumption that humans are consistent, and the stronger assumption that we can prove our consistency. Informal arguments are given to support these two premises. The informal arguments will not suffice as proof and, subsequently, the Gödel-arguments fail.

In 1961 Lucas published the widely known *Minds, Machines and Gödel* (*MMG*) in which he claimed that it is *impossible* for machines to replicate the human mind because of Gödel-1 (Lucas 1961). Penrose joined Lucas in 1989 with his book *The Emperor's New Mind* (*ENM*) with an argument of the same form (Penrose 1989). The two arguments are often seen as one and are therefore together called the "Lucas-Penrose argument", and it is the first Gödel-argument.

The Lucas-Penrose argument has been subject to various critiques and among the critics the consensus is that it fails to refute CTM (Benacerraf 1967) (Chalmers 1995) (LaForte *et al.* 1998) (Lewis 1969; and 1979) (Franzén 2005) (Bringsjord & Xiao 2000). Lucas responded to some of his critics (1968; 1970; and 1984), but the required assumption of human consistency remains. Penrose developed a new and stronger Gödel-argument that avoids one of the premises in his new book *Shadows of the Mind* (*SOTM*) (Penrose 1994). The new argument will be called the "Penrose-argument".

The Penrose-argument has been thoroughly discussed by various philosophers, mathematicians, and computer scientists. Among the critics are David Chalmers (1995), Per Lindström (2001; and 2006), Solomon Feferman (1995), LaForte *et al.* (1998), and Bringsjord and Xiao (2000). The critics of Lucas and Penrose are not necessarily supporters of CTM, rather, they believe the arguments put forth lack what is needed to refute CTM. This paper will inquire into the objections made by Chalmers, Lindström and Bringsjord and Xiao.

In 1996 Penrose published a paper named *Beyond the Doubting of a Shadow* in which he responds to some of the objections that had thus far been made against the Lucas-Penrose argument and the Penrose-argument, including Chalmers' objection (Penrose 1996).

The paper will begin with a theory section providing explanations for relevant terms, theorems and concepts that will be useful in understanding the debate. The paper will not provide proofs for the theorems (except for Penrose's version of the proof of undecidability of the halting problem in 2.4, which will be called the "Turing-proof"), instead there will be basic explanations which will aid in understanding their implications in what is later being discussed. After the theory section the paper inquire into the actual debate between Lucas and Penrose and their critics.

The Lucas-Penrose argument is introduced informally using Penrose's Turing-proof and Gödel-1,

[1] Also known as "Mechanism" or "Computationalism".

[2] Other arguments against Lucas' and Penrose's Gödel-arguments have been concerning technical issues and ambiguous use of terms (Feferman 1995) (LaForte *et al.* 1998) and a problematic idealization of humans (Coder 1969) (Boyer 1983).

and a formalization of the argument is then provided. Next, Lucas' original publication and some of its main critiques by the philosophers Paul Benacerraf (1967) and David Lewis (1969; and 1979) will be referred to. The debates between Lucas and his critics are supplemented with a discussion of the arguments. Finishing the Lucas-Penrose argument section there will be a concluding discussion as to the efficacy of the argument. All the discussions, conclusions and tables in the paper are solely made by me. If I use material by another author in this section, it will be referred to.

The new Penrose-argument will then be introduced with an informal description and a formalization. There will be an added explanation as to how it is improved from the Lucas-Penrose argument. Succeeding the explanation of the new argument an objection made by Chalmers (1995) will be cited. The objection is responded to by both Penrose (1996) and Lindström (2001). Subsequently, there will be a discussion of Chalmers' objection. Two final objections against the Penrose-argument by Lindström (2001) and Bringsjord and Xiao (2000) are then detailed. After presenting the objections against the Penrose-argument, a conclusion to the argument is given. The paper will then end with the main points of the paper restated and a conclusion to the Gödel-arguments stated.

2. Theory

2.1 Formal languages, formal systems, and interpretations

A *formal language* is mainly used in computer science, mathematics, and logic. It consists of a finite set of symbols (an *alphabet*) and a *syntax*. A combination of symbols from the language's alphabet constitutes a *string*[3]. A formal language that allows for *any* combination of symbols would rarely be found useful. Therefore, a formal language usually follows a *syntax*. The syntax is a set of rules deciding which combinations of symbols are allowed. Strings that are combined in accordance with the language's syntax are called *well-formed formulas* (wffs).

A *formal system* is a subset of a formal language and is constructed following certain *rules of formation*. The rules of formation are statements about the symbols and syntax. Further, it contains a *deductive system* which is made up of *rules of inference* or *axioms*, or both. The stipulated axioms need to be decidable – an algorithm deciding which formulae are axioms and which are not is required. Such a system is said to be "recursively axiomatizable". (Raatikainen 2021)

Proceeding from our axioms we can use the system's rules of inference to gain further theorems in the system. The theorems are either axioms or derivable from the axioms by the rules of inference. The theorems are made up of the symbols of the language and they are *wffs*. This does not imply that all wffs of a system are theorems in the system, since there are wffs in a language you could come up with that are not deducible from the axioms and rules of inference. "F⊢A" says that A is derivable in F, and "F⊬A" says that A is *not* derivable in F. (Britannica, 2012) (Wang & Schagrin, 2011)

The formal system is said to be *complete*[4] if any formula, or its negation, in a language is decidable (provable) within the system. The formal system is said to be *consistent* if there is no formula *and* its negation decidable in the system. An inconsistent formal system is practically useless since any sentence follows a contradiction (Raatikainen 2021).[5] Consistency is a minimal *soundness property* (Franzén, p. 21).

A formal language is wholly syntactic and for it to get a semantic meaning we need a semantical

[3] Or "sentence", "formula".

[4] More precisely *negation complete*. (Franzén 2005 p. 27)

[5] This is the "Principle of explosion". Ex falso quodlibet – "from contradiction anything follows". (Ex falso quodlibet, *Oxford Reference*)

interpretation of it. The way of interpreting a formal language is by model theory. Models (or "interpretations") of a language are set-theoretic structures that interpret the language's symbols. If a wff *S* is *true* in certain interpretation (model) *I*, it is said that the interpretation *I satisfies S*. This is written as "$I \vDash S$". Further, a variable assignment *g* that satisfies *S* in a certain interpretation *I* can be written as "$I \vDash S(g)$" (Barker-Plummer *et al.* 2009, p. 519). When a formula of a language is true under a certain interpretation it is said to be *sound*. A sound argument consists of true premises that always yield true conclusions – there exists no model in which the premises are true, and the conclusion is false, *and* the premises are in fact *true*[6]. (Wilfrid 2020) (Shapiro & Teresa 2021)

This definition of soundness differs from its use by Penrose. Penrose's "soundness" is in terms of arithmetic soundness (in the arguments cited in this paper) and refers to the strongest soundness property. In this case, a system is *sound* if every arithmetical theorem of the system is true. (Franzén 2005, p. 21)

2.1.1 Peano arithmetic

There are five axioms of additional importance – the Peano axioms. The Peano axioms concern natural numbers, and in our debate it is generally assumed that the discussed machines are at least capable of the operations the Peano axioms allow for. A certain capacity is required of a system for Gödel-1 to apply, and the Peano axioms provide with said capacity. A system containing the (interpreted in its language) Peano axioms is said to be capable of *Peano arithmetic* (PA). Below are the five axioms, courtesy of W. L. Hosch of the *Encyclopedia Britannica* (2010):

1. Zero is a natural number.
2. Every natural number has a successor in the natural numbers.
3. Zero is not the successor of a natural number.
4. If the successor of two natural numbers is the same, then the two original numbers are the same.
5. If a set contains zero and the successor of every number is in the set, then the set contains the natural numbers.

2.2 Turing Machines and the Church-Turing thesis

A Turing machine is an idealized model of computation that is aimed to uncover what the limits of computations are. It is an abstract machine, i.e., it is a theoretical model of a computer system (Macura, n.d.). It is idealized since it has a (potentially) infinite memory, it is not susceptible to any hardware failures, and it is not limited by a finite battery. A Turing machine can be said to be a formal model of a modern general-purpose computer and, accordingly, can be understood as the basis for our computers today. To solve certain problems of uncomputability, the Universal Turing machine was created. The Universal Turing machine is a Turing machine capable of simulating *any* other Turing machine's computations. A sketch of a Universal Turing machine is given at the end of section 2.2 in Figure 1. (De Mol, 2019)

The Turing machine works by reading its input from a (potentially infinite) tape. Along the tape are individual "boxes" that are either blank or filled with a **1** or a **0**. A reading device examines *one* box and its value at a time, starting from the left side of the tape. At any moment, the machine will be in an internal state that will determine the action depending on the value of the box examined. Being in a certain internal state scanning a specific box, the machine's program will order it to first (i) decide whether any change is to be made to the box's value, then it will (ii) determine, depending on the examined value, its new internal state, and finally it will decide (iii) what move the reading device is to make along the tape; move one step to the right, one step to the left, or one step to the right and halt. When (and/or if) the action of the machine stops, the boxes to the left of the reading device and their values will display the resulting answer to the finished computation. A computation is the

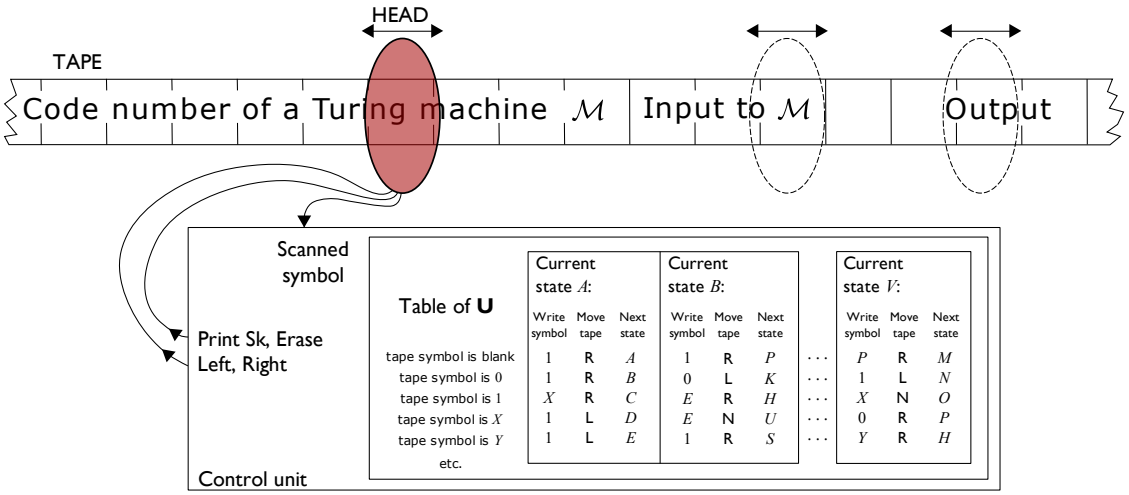
[6] True in the sense of "apples are fruits" or "the moon orbits the earth". They are true in the "real world".

action of a Turing machine. (Penrose 1994, Appendix A)

The *Church-Turing*[7] *thesis* states that *all* computable problems are *Turing-computable*. If a problem is not computable by a Turing machine, the thesis implies that it is not computable by any means at all. A Turing machine is said to give the most precise definition of what a computation is, and the Church-Turing thesis is well agreed upon by computer scientists today. (De Mol, 2019)

A quick note on top-down and bottom-up processing is solicited, as they are relevant to the debate on a grander scale. A computation is called "top-down" if it is an operation based on some fixed and well-defined rules that are always applied, i.e., the results of possible computations in the system are always determined according to the system's rules and will not change. On the other hand, "bottom-up" processing will see "improvements" over time, as the system memorizes earlier computations and adapts accordingly for better future results. It can be said to be "learning by doing". Systems can utilize both top-down and bottom-up processing. Speaking of *artificial intelligence* today it is usual to have *machine learning* and, subsequently, *artificial neural networks* in mind. Such systems are bottom-up. (Penrose 1994, §1.5)

Figure 1: A Universal Turing machine.



By Cbuckley - Own work, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=3097974

2.3 Decidability and arithmetization

We can now further expand on formal languages and relevant computability theory. A set of strings *E* is *recursively enumerable* (also called *semi-decidable* or *computably enumerable*) if there is a mechanical procedure that can generate them. Then, if every string of a formal language is enumerable by a Turing machine it is said to be recursively enumerable. The set can exist of any mathematical object that is representable by a string, such as the natural numbers. (Franzén 2005, p. 61–63)

In contrast, a set of strings *E* is *computably decidable* (also called *decidable* or *computable*) if a Turing computer can decide whether any string of symbols *s* is in *E* or not. Viz., if *s* is a Turing machine's input, the machine will compute and then output whether "yes" it is in *E*, or "no" it is not. (Franzén 2005, p. 64–65)

With a notion of formal systems and Turing machines we may now understand how they work together. The connection is made by a part of Gödel's proof, namely *arithmetization* (also called

[7] "Church" after the American mathematician Alonzo Church. (De Mol 2019)

Gödel numbering). Arithmetization of a formal system is the way of assigning natural numbers to a formal system's components – its terms, formulae, and proofs. Everything in the formal system will correspond to its encoded counterpart. The corresponding number is called a "Gödel number". In this way a machine may be an interpreted formal system, operating on the Gödel numbers that are arithmetized from the system's components. The machine's computations are operating on Gödel numbers corresponding to the wffs in a formal system. Any recursively enumerable formal system can therefore have a corresponding Turing machine, and vice versa. Machines (the ones in our debate) and formal systems are therefore interconnected, and throughout the paper "machine" and "formal system" will be used interchangeably. It will be made explicit when a machine that is not based on a formal system is mentioned. (Raatikainen 2021)

2.4 Gödel-1

Gödel's first incompleteness theorem (informally) states that no *consistent* formal system *F* that is capable of PA is complete, since there exists a Gödel formula within the system that is undecidable. Therefore, no formal system can be consistent *and* complete.

An easy way to understand the Gödel formula is by the (informal) sentence "This formula is not provable in the system". This sentence is closely related to the Liar paradox, which can be written as – "This sentence is not false". Gödel-1 may be described as follows:

If *F* is consistent → G_F (If *F* is consistent there is a Gödel formula G_F in *F*),
If *F* is consistent → $F \nvdash G_F$ (If *F* is consistent, *F* cannot decide G_F),

where *F* is a formal system capable of PA[8], and G_F is *F*'s Gödel formula. The Gödel formula is a wff and has a Gödel number. (Raatikainen 2021)

2.5 The halting problem

In *SOTM* Penrose uses a theorem by Turing for the Lucas-Penrose argument instead of the Gödel-1 theorem. Penrose gives Turing's theorem which states that the *halting problem* is undecidable. The halting problem has to do with determining whether a Turing machine's computation will halt or not (De Mol 2019). The proof for the "unsolvability" of the halting problem was used by Turing to prove his Undecidability Theorem[9] (Franzén 2005, p. 68). Using this proof instead of Gödel-1 will not conflict with the Lucas-Penrose main conclusion. To cite Geoffrey LaForte, Patrick J. Hayes, and Kenneth M. Ford –

"…every 'Gödelizing' argument using the unsolvability of the halting problem has a mirror image using Gödel's actual theorem. On the other hand, notions like consistency and truth are important for both our discussion and Penrose's, and these notions are explicitly involved only in the original incompleteness theorem. To keep the discussion manageable, we will follow Penrose by using the unsolvability of the halting problem in the most technical parts of our discussion, while talking as though we have been using Godel's theorem the rest of the time." (1998, p. 267)

I will follow the same structure. The Turing-proof will be alluded to when it is conductive. In both the Turing-proof and Gödel-1 the *diagonalization* argument by logician George Cantor plays a major role (Franzén 2005, p. 70).

The proof provided by Penrose is in terms of *soundness,* while Gödel-1 speaks of *consistency*. LaForte *et al*. points out that Penrose conflates his use of "sound" (1998, p. 270), however, in this case it is implicit that arithmetic soundness is used (a sound computation only produces *true*

[8] PA is more than sufficient. The weakest required arithmetic for Gödel-1 to apply to a system is Robinson arithmetic. (Raatikainen 2021)

[9] The Undecidability theorem shows that "There are computably enumerable sets which are not computably decidable". (Franzén 2005, p. 68)

statements). "If *A* [the computation soon to be described] does not in fact give us wrong answers, we say that *A* is *sound*" (Penrose 1994, p. 73). Soundness therefore implies consistency. Consistency is a weaker soundness property than arithmetic soundness. Both terms will be used in the debate, with "sound" appearing more often in the discussions concerning the Penrose-argument. The following proof is in terms of a Turing machine as the computational device.

Consider a computation that acts on *any* natural number *n*. We denote this family of computations by $C(n)$. What it means for a computation to act on a natural number can be explained with an example. Consider this computation:

> Find an odd number that is the sum of *n* even numbers.

We see that the computation above will never find any such odd number that is the sum of *n* even numbers. The computation will sift through the infinite natural numbers, i.e., the computation will not stop.

Can we construct a procedure that decides whether a never-ending computation in fact never ends? Consider a sound computation *A*. *A*'s job is to determine whether a $C(n)$ stops or not. If *A* terminates, we know that $C(n)$ never stops. Of course, the $C(n)$ computations are not all the same. To differentiate different $C(n)$ computations we assign them numbers – $C_0(n)$, $C_1(n)$, $C_2(n)$… etc. Let *q* denote the assigned number of $C(n)$, so that $C_q(n)$. $C_q(n)$ is the *q*th computation acting on the natural number *n*. *A* will be any *sound* set of computational rules, operating on the numbers *q* and *n* to determine whether $C_q(n)$ stops or not. If *A* terminates, $C_q(n)$ is a computation that stops. The computation *A* performs one the numbers *q* and *n* and can therefore be denoted as $A(q, n)$.Then we have:

(1) If $A(q, n)$ stops, $C_q(n)$ does not stop.

Now we begin the first step of Cantor's "diagonal slash", which is also utilized in Gödel-1. We put *q* equal to *n*. This gives us:

(2) If $A(n, n)$ stops, $C_n(n)$ does not stop.

A is now only operating on *one* number, *n*, which will be one of *all* the possible *q*th computations operating on *n*. Consider that it is the computation C_k. Then it is true that:

(3) If $A(n, n) = C_k(n)$.

Now we proceed to the second step of the diagonal slash. Consider that *n* and *k* share the same value. Then:

(4) If $A(k, k) = C_k(k)$.

From (2), with $n = k$, we get:

(5) If $A(k, k)$ stops, $C_k(k)$ does not stop.

And now, considering (4).

(6) If $C_k(k)$ stops, $C_k(k)$ does not stop.

Our conclusion must be that if we *know* the computation *A* to be sound, then we know the computation $C_k(k)$ does *not* stop. If it would stop, it does in fact not stop, according to (6). However, our computation *A* that we are using for seeing whether $C_k(k)$ stops or not does not stop either, since we know them to be one and the same according to (4). This shows that the computation $A(k, k)$ is not able to ascertain whether $C_k(k)$ stops or not, since itself never stops. *We* know something that *A* does not (that $C_k(k)$ never stops), if we *know* that *A* is sound. (Penrose 1994, §2.5)

This is Penrose's version of Turing-proof, and it will soon show how it is relevant for the Lucas-Penrose argument. "We deduce that no knowably sound set of computational rules (such as *A*) can

ever suffice for ascertaining that computations do not stop, since there are some non-stopping computations (such as $C_k(k)$) that must elude these rules." (Penrose 1994, p. 75).

We further deduce that no *sound* formal system capable of PA can fully capture *all* truths pertaining to natural numbers, since there exists a statement about natural numbers that is undecidable; no *complete* and *sound* axiomatization of natural numbers is available. (Penrose 1994, §2.5)

2.6 Gödel-2

Gödel's second incompleteness theorem is a corollary to the first. A brief and informal version of it will suffice for understanding its later use. The theorem (informally) states that no *consistent* formal system *F* that is capable of PA can prove its own consistency (it cannot contain a formula which states its consistency). The implications of Gödel-2 are just as important for the debate as Gödel-1. Gödel-2 will later be used against Lucas' and Penrose's arguments. Gödel-2 can be explained as:

If *F* is consistent $\rightarrow F \nvdash \text{Con}(F)$ (If *F* is consistent, then *F* cannot prove that *F* is consistent),

where *F* is a formal system capable of PA. (Raatikainen 2021)

3. The Lucas-Penrose argument

3.1 The Lucas-Penrose argument

The Lucas-Penrose argument works by *reductio ad absurdum*[10]. If a machine operating on a consistent formal system is to be able to replicate a human mind it must be able to in principle perform what a human mind in principle is able to. Since it cannot see the truth of its Gödel formula (or that the computation $C_k(k)$ never halts), which Lucas and Penrose believe the human mind can, CTM must fail. For every consistent formal system, we can see that it has a *true* formula that in the system is *undecidable*. Lucas and Penrose believe we can in principle prove something that a machine in principle cannot, which implies that no machine (i.e., formal system) can in principle emulate a human mind. The Lucas-Penrose argument will now be explained using the Turing-proof. (Lucas 1961) (Penrose 1994, §2.5)

Imagine that the computational procedure *A* (as given in 2.3) encapsulates *all* the humanly available mathematical reasoning for ascertaining whether a computation stops or not. If *A* stops, it shows that our human mathematical understanding is sufficient for ascertaining that the pertaining computation ($C_k(k)$) does not stop. Since *A* corresponds to an idealized human, we conclude that it is sound, and that the computation $C_k(k)$ does not stop. However, as we have seen, *A* cannot ascertain this. Since *we* know that the pertaining computation does not stop, *we* know something that a *sound* computation *A* cannot in principle know. The conclusion must then be that a sound computation cannot encapsulate all humanly available mathematical reasoning, considering that *we* can deduce something *A* cannot (that $C_k(k)$ in fact does not stop). Penrose formulates the conclusion:

G – "Human mathematicians are not using a knowably sound algorithm in order to ascertain mathematical truth." (1994, p. 76) (Penrose 1994, §2.5)

In *SOTM* Penrose uses this proof as the argument; however, the same conclusion was drawn by Penrose in 1989 in *ENM* using Gödel-1 (1989, p. 416–418).

The main conclusion of the Lucas-Penrose argument is that the human mind cannot be a formal system (i.e., a machine) since we can ascertain something that a consistent formal system *never*

[10] Proof by contradiction – assuming something to be true to show that it leads to absurdity/contradiction, and that therefore the opposite is true. (Britannica 2017)

could. Our reasoning must contain something that is not computable, and we are therefore fundamentally different. (Lucas 1961) (Penrose 1989; and 1994)

3.1.1 The Lucas-Penrose argument formalized

Let *F* be a formal system. Con(*E*) is a formalization for "*E* is consistent" and T(*E*) is a formalization for "*E* is true". $\boldsymbol{M}$ is the (in principle available) human mathematical reasoning. $F_{\boldsymbol{M}}$ means that *F* is based on $\boldsymbol{M}$ (practically $F = \boldsymbol{M}$). $G_{F_{\boldsymbol{M}}}$ is *F*'s Gödel formula. It is assumed that $F_{\boldsymbol{M}}$ is capable of PA. The contradiction is derived by us ($\boldsymbol{M}$). T(*E*) means that *we* conclude that the formula *E* is true (in $\boldsymbol{M}$). The Lucas-Penrose argument may be formalized in the following way, see Table 1:

Table 1: Lucas-Penrose argument formalized.[11]

Con($\boldsymbol{M}$)	Premise ($\boldsymbol{M}$ is consistent)	(1)
Con($\boldsymbol{M}$) → Con($F_{\boldsymbol{M}}$)	Premise (If $\boldsymbol{M}$ is consistent, $F_{\boldsymbol{M}}$ is consistent)	(2)
$F_{\boldsymbol{M}}$ → T($G_{F_{\boldsymbol{M}}}$)	Premise (If *F* is based on $\boldsymbol{M}$, we know $G_{F_{\boldsymbol{M}}}$ is true)	(3)
Con($F_{\boldsymbol{M}}$) → $F_{\boldsymbol{M}} \nvdash G_{F_{\boldsymbol{M}}}$	Premise (If $F_{\boldsymbol{M}}$ is consistent, $G_{F_{\boldsymbol{M}}}$ is undecidable in $F_{\boldsymbol{M}}$ (Gödel-1))	(4)
$F_{\boldsymbol{M}} \nvdash G_{F_{\boldsymbol{M}}}$ → ~T($G_{F_{\boldsymbol{M}}}$)	Premise (If (4), $G_{F_{\boldsymbol{M}}}$ cannot be true as it is undecidable)	(5)
⊥	Contradiction (T($G_{F_{\boldsymbol{M}}}$) and ~T($G_{F_{\boldsymbol{M}}}$))	(6)
~$F_{\boldsymbol{M}}$	Conclusion (*F* cannot be based on $\boldsymbol{M}$, $F \neq \boldsymbol{M}$)	(7)

Premise 1: our ($\boldsymbol{M}$) reasoning is consistent.
Premise 2: if $F = \boldsymbol{M}$ ($F_{\boldsymbol{M}}$), $F_{\boldsymbol{M}}$ is also consistent.
Premise 3: we can see that the system's Gödel formula is true.
Premise 4: however, a consistent formal system cannot decide its Gödel formula according to Gödel 1.
Premise 5: it is implied that $F_{\boldsymbol{M}}$ cannot see that $G_{F_{\boldsymbol{M}}}$ is true if $G_{F_{\boldsymbol{M}}}$ is undecidable.
Premise 6: we know $G_{F_{\boldsymbol{M}}}$ is true, but this is impossible for a formal system.
Premise 7: we cannot be a formal system. $F \neq \boldsymbol{M}$.

Since we know that we (practically) are $F_{\boldsymbol{M}}$ we know that $G_{F_{\boldsymbol{M}}}$ must be true. However, no consistent formal system can contain its Gödel formula. Since *F* is *any* formal system, it follows that *no* formal system can be based (contain) the (in principle available) human mathematical reasoning. If that is the case, we cannot in principle be formal systems. Conclusion: a formal system cannot emulate a human mind.[12] (Lucas 1961) (Penrose 1989; and 1994)

3.2 Minds, Machines and Gödel – John Lucas

Lucas introduces the Gödel formula as "This formula is unprovable-in-the-system" (1961, p. 1). The formula is undecidable within a consistent formal system adequate for PA. *We* outside of the system can see that it must be true (if we know the system is consistent), precisely because it is unprovable in the system. Lucas calls it the "Achilles' heel of the cybernetical machine." (1961, p. 5). A consequence of Gödel-1 is that either the system is consistent but incomplete, since the Gödel formula is undecidable in consistent systems, or the system is complete but inconsistent. Subsequently, Lucas forms the conclusion that "It follows that no machine can be a complete or adequate model of the mind, that minds are essentially different from machines." (1961, p. 2).

After making the anti-CTM conclusion explicit Lucas goes on to put up a defense against some of the possible counter arguments. The first objection Lucas responds to involves creating new machines capable of deciding earlier machines' Gödel formulae. This can be done. However, even if a new machine M_2 can decide the first machine M_1's Gödel formula, M_2 will in its turn (assuming it is operating on a consistent formal system and is capable of PA) have its own Gödel formula that is constructed according to *its* formal system. M_1's Gödel formula remains undecidable for M_1, M_2 may prove it but will nevertheless have its own undecidable Gödel formula. And, naturally, creating a

[11] This is my own formalization. Other formalizations are also possible.

[12] Understanding *F* as the computation *A,* and "Consistent" as "Sound", will yield the same conclusion.

consistent machine M_3 adequate for proving M_2's formula will result in M_3's own undecidable formula . We see that constructing new consistent machines is not a solution for avoiding the Gödel formula, as each machine will have its own undecidable formula. (Lucas 1961, p. 5–6)

The second objection Lucas responds to regards what he calls a "Gödelizing operator". We add to a machine's formal system an additional rule of inference that adds the Gödel formula for the system as a theorem (in effect making it a new system), and then a theorem for the new system with a new Gödel formula, and then a theorem for the new system… *ad infinitum*. For each new Gödel formula, the system will have a corresponding Gödelizing operation stemming from the addition to its formal system's rules of inference that makes the formula decidable (and operation computable). The problem with this, Lucas writes, is that the formal system with a Gödelizing operator will itself be susceptible to a Gödel formula. It is true that the Gödelizing operator will let the machine proceed through an infinite amount of Gödel formulae, but to this infinity another formula is added to the system containing the operator. The new infinity is larger than the initial infinity of Gödel formulae and contains an undecidable Gödel formula. The formal rules in the system specifying the axiom scheme needed for the Gödelizing operator will, with the rest of the formal system, have a Gödel formula. (Lucas 1961, p. 6–7)

The third objection is made by Turing himself. Turing points out that even if there is an undecidable formula in consistent systems it does not amount to anything much in terms of humans being "better" than machines. It is one single aspect of human "superiority", but there are other ways machines outclass humans. Lucas agrees with this, but it is not an objection to Lucas' anti-CTM proposal. It is true that there are ways in which a machine is superior to humans, certainly concerning mathematics, but this is not what Lucas is arguing for. Lucas says that a machine and the human mind are fundamentally *different*, since the discrepancy caused by the Gödel formula is always going to exist. (Lucas 1961, p. 7–8)

Now the fourth objection. Gödel's incompleteness theorems apply to *formal* systems which are capable only of deductive methods of proof. Is it fair to compare such a machine with a human, who in addition to deduction uses other methods of inference? A machine operating under formal rules is *inevitably* incomplete if it is consistent. It might be fair to say that a machine only capable of logical deduction is fundamentally different from the human mind, but what about machines accessing the same methods as humans? Lucas responds by saying that a machine capable of producing theorems that are not deduced from its axioms according to its formal rules would be unsound, even if it is not inconsistent. (Lucas 1961, p. 8–9)

Another problem that follows is Gödel-2. We need to *know* that the examined formal system *is* consistent to know that it has a Gödel formula. However, Gödel-2 precludes us from obtaining a formal proof of such consistency. Even if a system has been consistent thus far, there is no way of knowing that the system might show inconsistencies in the future. Lucas concludes that "At best we can say that the machine is consistent, provided we are. But by what right can we do this?" (1961, p. 10). The Lucas-Penrose argument assumes that we are consistent, that we can show that we are (since we must know that *F* and *A* are consistent), *and* that we can see the truth of our Gödel formulae. There must then exist some way that humans escape Gödel-2's implications. (Lucas 1961, p. 10)

First, Lucas responds by saying that we *sometimes* are inconsistent, but that this inconsistency is not tantamount to the inconsistency that we are referring to when speaking of formal systems. *Our* inconsistencies are merely "mistakes", analogous to a machine's hardware error. He believes the important thing to recognize is that we correct these mistakes and logical inconsistencies as soon as we notice them. No human *knowingly* holds two contradictory propositions. If we truly were inconsistent, we would not correct these mistakes, but rather keep them and repeat them. Humans do discriminate between false and true propositions, and a human not doing this is, Lucas points out, is said to have "'lost his mind'" (1961, p. 11). In short, a human mind does not adhere to the earlier mentioned principle of explosion. Either way, it is not such a human mind we are comparing the machines with, rather, we are comparing idealized machines with idealized humans and their fundamental limits. If we contrast such human fallibility and a machine with the same type of

fallibility but self-correctability, the machine will still have its undecidable Gödel formula. Lucas believes the human mind is consistent, and that it can assert its own consistency. It is this type of transcending self-conscious consistency, unscathed by Gödel-2, that is ultimately needed for the Gödel-arguments to succeed. (Lucas 1961, p. 10–11)

In favor of CTM, Lucas proposes a machine that is normally consistent, except for when it comes to proving the Gödel formula. In short, Lucas' conclusion to this argument is that it would not be viable. The machine would lose its sense of logic, be arbitrary and irrational. It would not be comparable to the human mind. Once a machine has shown inconsistency it *is* inconsistent, and the principle of explosion applies. (Lucas 1961, p. 12–13)

Lucas further responds to the problem of consistency stating that the only thing Gödel-2 proves is that a human mind cannot formally prove a formal system's Gödel formula from *within* the system. However, Lucas sees no objections to proving the consistency of a system from outside the system with *informal* arguments. "Such informal arguments will not be able to be completely formalized…" (Lucas 1961, p. 15). (Lucas 1961, p. 14–15)

Lucas concludes that the self-referring nature of the Gödel formula is what makes it undecidable for machines, but decidable for man. It requires a certain self-awareness to prove it, since the formula is a statement about oneself. Lucas likens introducing a machine to its Gödel formula with "...asking it to be self-conscious..." (Lucas 1961, p. 15). Self-consciousness is the "act" of knowing that you know, and knowing that you know that you know... It gives rise to a certain self-referential ability. *We* are capable of this, *we* are self-conscious, and it is therefore *we* can prove the formula, unlike the machines (while also being consistent). "If the mechanist can devise a model that I cannot find a fault with, his thesis is established: if he cannot, then it is not proven: and since---as it turns out-he necessarily cannot, then it is refuted." (Lucas 1961, p. 8). (Lucas 1961, p. 15–16)

3.3 Objections by Paul Benacerraf

Before inquiring into the article's main objections Benacerraf mentions the fact that Lucas' argument presupposes that the computationalist's machine corresponds to a Turing machine. Other computational devices also exist. Refuting CTM then requires more than just refuting Turing machines. Benacerraf does not further discuss this objection. (Benacerraf 1967, p. 13–14)

Now to Benacerraf's main objections. Benacerraf points out that Lucas' argument is ambiguous. What a machine cannot prove *formally* with its deductive system Lucas allows himself to prove *informally*. We saw that Lucas sees no problems with this. However, in the same way it is impossible for a consistent formal system to derive a proof of its Gödel formula, it is just as impossible for Lucas to do so using the same deductive system as the machine. Letting Lucas prove the machine's Gödel formula informally from outside the system is "cheating". Benacerraf explores the idea of a consistent formal system acquiring an informal conviction of the truth of its Gödel formula because of Gödel-1 (implying that it believes it is consistent). We cannot accept this informal conviction as proof. A formal system "convincing" itself of proofs of its formulae cannot be said to be consistent (as is also mentioned by Lucas (1961, p. 12–14)). Lucas' informal arguments for the consistency of a system from outside the system lacks the same formality to count as an actual proof for consistency. Lucas' informal "belief" of the truth of a system's Gödel formula should in principle be available for a machine as well. A machine may also be aware of Gödel-1, and hence informally "believe" it has a true, yet undecidable in the system, formula. (Benacerraf 1976, p. 19–20)

To make the argument less ambiguous, we can interpret Lucas as saying that he can prove a machine's Gödel formula from *within* using the same deductive system. However, there is no possible way Lucas could in fact do this. As Benacerraf writes, "…*if* he can do this, he is not a machine, and, if he is not a machine, then Gödel's theorems do not preclude his being able to do it." (1967, p. 21). (Benacerraf 1967, p. 20–21)

The second main objection is concerning consistency. For the Lucas-Penrose argument to succeed we need to *know* that the formal system we are examining is *consistent* (and that the computation *A* is sound). Simply knowing about Gödel-1 does not imply that a system is consistent, and since a

consistent formal system cannot assert its own consistency (Gödel-2), we cannot formally prove its consistency. How are we then to make sure of a formal system's consistency (or a computation's soundness)? Both Penrose and Lucas depend on the assumption of such knowledge. Torkel Franzén comments on this in *Gödel's Theorem: An Incomplete Guide to Its Use and Abuse* – "If the human mind did have the ability to determine the consistency of any consistent formal system, this would certainly mean that the human mind surpasses any computer, but there is no reason whatever to believe this to be the case." (2005, p. 55). (Benacerraf 1967, p. 21–22)

Further, Lucas claims that CTM is false if he can find the Gödel formula in *any* machine a computationalist produces (1961, p. 8). Even if Lucas can do so, other machines can informally see the proof of other machines' Gödel formulae as well. Lucas is not superior because of this ability. Benacerraf also points out that it is not always conceivable that Lucas *in fact* is able to find the flaw in each machine. (Benacerraf 1967, p. 22–23)

Benacerraf's final main objection is that if we are Turing machines we cannot know which one we are (on which program we are running). "If I am a Turing machine not only can I not ascertain which one, but neither can I ascertain any instantiation of the machine that I happen to be that it is an instantiation of that machine." (Benacerraf 1967, p. 29). This fact seems to be well accepted among the pertinent literature, and a proof for this will therefore not be necessary.[13] This implies that if we are Turing machines, we cannot know if we are consistent or inconsistent (since we do not know our code, and Gödel-2). Then, there is a possibility that we are inconsistent machines capable of "proving" (from the principle of explosion) our Gödel formulae. More importantly – if I cannot know how I am formalized, I need to be able to prove the consistency of the formal system *F* or soundness of computation *A without* knowing that they correspond to me. The system's consistency must be established for the contradiction to rise. (Benacerraf 1967, p. 29–30)

3.3.1 Lucas's rejoinder to Benacerraf

Lucas responds to Benacerraf by saying that the argument is misunderstood. Lucas' argument should be taken as a dialectical one, a "schema of disproof" (Lucas 1968 p. 145). The computationalist is allowed to put forth *any* machine for Lucas to examine, and if he can find its Gödel formula (and see its truth) CTM fails, according to Lucas. Even if another machine could out-Gödel the same machines as Lucas, he could out-Gödel any machine that is put forth, and Lucas believes that is what counts. The argument rests on the requirement that only *one* machine is put forth and that it is fully specified. (Lucas 1968, p. 145–146)

Lucas then responds to the objection of the claim that Lucas can informally see the truth of a system's Gödel formula while the machine is expected to have a formal proof, and that he therefore equivocates on the word "proof". Lucas argues that truth is more than provability-in-a-given-system, i.e., more than formal proofs in given proof systems. He does not claim to have an *exact* notion of truth, but still believes to *know* truth for what it is. In line with his response to seeing the soundness of a formal system (Lucas 1961, p. 14–15), he believes informal arguments can be utilized in cases such as these. (Lucas 1968, p. 147–149)

The objection regarding not knowing our code if we in fact are machines is responded to by referring to the dialectical form of Lucas' argument. It is not necessary to know whether the machine's mechanisms correspond to Lucas', Lucas only needs the machine's mechanisms specified. Lucas reiterates the fact that it is not a matter of a proof sequence (such as my formalization of the Lucas-Penrose argument), but a matter of dialectic. Furthermore, Lucas claims Benacerraf's argument is guilty of inconsistency – "Benacerraf is claiming that the man is a machine, although for every particular machine he could be we could show that he is not that one." (Lucas 1968, p. 151). Hence, Lucas believes the argument is in fact in favor of anti-CTM. Lucas insists that Benacerraf, to avoid inconsistencies, should instead shift focus to the argument of Lucas not being able to know

[13] For a more complete discussion of this, I refer to *A machine that knows its code* by Samuel A. Alexander (2013).

whether a putative system is consistent or not. (Benacerraf does in fact mention this problem in (1967, p. 21–22)). (Lucas 1968, p. 150–152)

Given the dialectical argument, Lucas receives the specifications of a machine the computationalist claims to correspond to Lucas and then calculates its Gödel formula. Lucas then asks the computationalist whether the machine (i.e., formal system) can decide its Gödel formula. If he says yes, Lucas knows it is *inconsistent*, and therefore not him. If he says no, Lucas knows that the machine is *consistent*, and therefore cannot prove its Gödel formula (unlike Lucas). Either answer, Lucas believes that it follows that he is not the machine. Hence the name of the article (*Satan Stultified*). This is how Lucas argues against not knowing whether a machine is consistent or not – he merely needs to ask the computationalist for a further specification. (Lucas 1968, p. 152)

Lucas then responds to what he believes is the final possibility of a refutation of his argument – are humans consistent? A computationalist might claim that we in fact are mere inconsistent machines capable of proving anything.[14] If we are consistent, we should not be able to formally claim consistency (Gödel-2). We already know how Lucas responds to this in *MMG*, and he uses the same arguments in this article. Lucas believes we are consistent since we are selective of what we hold as true and false. We consciously avoid inconsistencies and remove them when they are made aware of. Consistency is a decision we make, and our rationale lets us claim that we are rational and consistent. It is on this basis we can make future thoughts, knowing we are consistent, according to Lucas. There is no logical deduction for this claim of consistency, but we can know we are since we are selective. Lucas believes the other option is both nihilistic and contradictory. (Lucas 1968, p. 157–158)

3.3.2 Lucas and Benacerraf: discussion

Benacerraf mentions the possibility that a machine not corresponding to a Turing machine might be capable of being analogous to a human mind. If that is the case, it is not enough to refute only the Turing machine. This objection is worth further discussion, as it binds to the limits of computation. My response to this is that if we hold the Church-Turing thesis to be true, a Turing machine is in fact the best candidate to represent CTM. In the light of new technology (or new possible theories expanding our theoretical limits of computation), however, this might of course turn out to be false. Although, as of right now, CTM must accept a Turing machine as its best candidate if we hold the Church-Turing thesis to be true.

The case for Lucas not being capable in practice to prove the Gödel formula of any proposed machine does not really affect Lucas' main argument. The main point that Lucas is trying to make is that a machine cannot *in principle* emulate a human mind. This fact has been made clear in his papers.[15] There are other ways in which Lucas' argument fails (is he capable of *proving* the formula at all?), but since we are dealing with an *idealized* human[16], this is not one of them.

Lucas claims that the fact that the argument of not knowing which machine we are if we are machines is guilty of inconsistency. It is true that I cannot know which machine I am if I am a machine, and that no proposed code will suffice for me to accept it as mine. This does not imply that I am not a machine. The fact that a formal system cannot know itself does not imply that it is not a formal system. If Lucas is a machine and is presented with Lucas' code, he will reject it being his. This does not prove anything. The argument fails.

Lucas' way of avoiding having to prove the consistency of a system by asking the Computationalist a question regarding the machine is ingenious, and I have not seen any pertaining literature discuss this possible response. Lucas is allowed a full specification of the machine he is to examine – is it fair

[14] This argument was first mentioned by Hilary Putnam (Putnam 1960), before *MMG* was published.

[15] For instance, Lucas comments on this in his second rejoinder to Lewis – "I can in principle (my critics are often very charitable in speaking as though I could in practice, but let me revert to an ideal mind) calculate a Godel sentence for that machine - indeed infinitely many, depending on the Godel numbering scheme adopt". (1984, p. 189)

[16] This idealization may lead to further problems that are not discussed in this paper. (Coder 1969) (Boyer 1983)

to allow him the answer to the question whether the machine can prove its Gödel formula or not? It depends on if it is included in the *formal* description of the machine's deductive system that Lucas is given. It is also assumed that the computationalist knows the answer. Either way, the conclusion he draws once again rests on the assumption of human consistency. The implications he draws from the "no"-answer are incorrect. It must be assumed that a formal system that corresponds to us can prove its Gödel formula. There is no proof of this. A human, or other formal systems, can prove the Gödel formula from outside its system; this does *not* imply that we can prove a Gödel formula of a system from *within* the system. Lucas assumes that we are capable of this, and hence draws the conclusion that if the system cannot prove its formula it cannot correspond to us. The argument fails, as it rests on this assumption.

The final objection is that of knowing our consistency – a crucial part of the Lucas-Penrose argument no matter which form it takes (dialectic or not). Once again, it comes down to the same assumptions of human consistency. This assumption is made without any form of *actual* proof (we surely cannot accept the informal and intuitive arguments made by Lucas as proof). It is not legitimate to make such assumptions with formal implications based on informal arguments. This is enough for the whole argument to falter. I will further argue against this assumption in the conclusion to the Lucas-Penrose argument.

3.4 Objections by David Lewis

In 1969 the philosopher David Lewis put forth a counter argument to Lucas'. In the paper Lewis argues that unless Lucas can produce all "Lucas arithmetic" (the arithmetic Lucas can possibly produce) he cannot prove himself not to be a machine. Lucas needs to be able to verify *any* theorem of Lucas arithmetic as such. Since it is highly unlikely that Lucas in fact can do so, Lewis concludes that Lucas cannot prove his thesis. (Lewis 1969)

Shortly after, Lucas published a response (Lucas 1970) to Lewis in which he clarified his thesis, rendering Lewis' paper ineffective as a counter argument. In Lucas' response he claims that his argument is dialectical and dynamic, matching the response that was made to Benacerraf (Lucas 1968). It is only *when* a mechanist puts forth a machine that Lucas needs to show that there is a theorem he can see the truth of, unlike the machine. Therefore, it is only the part of Lucas arithmetic that is responsible for such a demonstration that is needed of Lucas. With Gödel-1 Lucas believes he will be able to show this.

With Lucas' response in mind, Lewis revised his objection (Lewis 1979). The new objection involves differentiating between Gödel formulae and a machine's output according to its input. The argument is related to the objection of having to know the putative system's consistency. Lewis argues that "To confuse the two sorts of Gödel sentences is a mistake." (1979, p. 376). Lucas' overall potential output must be separated from his output when accused of being a certain machine. We let O_L be Lucas' potential arithmetical output when not accused of being a machine and O_L^M be Lucas' output when accused of being a certain machine M. In the same way, a machine M's potential arithmetical output is O_M when not accused of being a machine and O_M^N when M is accused of being a machine N.

Lewis then grants Lucas three premises: (i) O_L is true (it is sound, implying consistency), (ii) O_L is capable of PA, and (iii) O_L^M consists of O_L plus the Gödel formula ϕ_M, which Lewis specifically defines as "expressing the consistency of M's arithmetical output" (1979, p. 374). In other words, it is not the Gödel formula related to Gödel-1. It is a formula related to Gödel-2, stating the consistency of a system. This premise lets L see the consistency of M, and thus prove the Gödel(-1) formula. We also grant his computational abilities a practical idealization (in line with the earlier mention of idealized machines) and a full specification of the machine he is accused of being. In other words, if he in theory can produce a proof for the Gödel formula of the machine that he is accused of, he has every tool needed to do so. This is in line with Lucas' original premises in *MMG*. Lucas believes that O_L is consistent and (in principle) capable of seeing the truth of any machine's Gödel formula that he is accused of being. (Lewis 1979, p. 373–375)

Lewis lets the accusation that Lucas is a machine M be true so that he can derive the usual proof by contradiction that the Lucas-Penrose argument employ. Then, $O_L = O_M$ and $O_L^M = O_M^M$. From any input into M the output will be recursively enumerable, therefore M is axiomatizable. We then let ϑ be the formal theory with theorems deducible from O_M^M. Conforming to Gödel-2, if ϑ contains a Gödel formula that expresses its own consistency; ϑ is inconsistent. ϑ do contain ϕ_M; ϕ_M is in O_L^M and therefore in O_M^M. Then ϑ must be inconsistent. This is the contradiction. ϑ contains ϕ_M which makes it *inconsistent*, while at the same time ϕ_M (stating the consistency of M) is true according to (i), which makes ϑ true and *consistent*. (Lewis 1979, p. 375–376)

However, the argument is incomplete according to Lewis. What is missing is a specification of whether ϕ_M states the consistency of O_M or O_M^M (the output of Lucas assumed to be M when not accused of being M, or when accused of being M). If it states the consistency of O_M, ϕ_M is true according to (i), and since we have O_M plus ϕ_M, we get O_M^M. However, since ϕ_M does not state the consistency of O_M^M (Lucas is being accused of being M, and therefore his output must be altered thereafter), a different system from O_M, the proof by contradiction fails.

The second case, in which ϕ_M states the consistency of O_M^M, ϕ_M also makes a statement about the consistency of ϑ (since the formal theory ϑ is O_M^M axiomatized). Since ϕ_M cannot belong to ϑ without ϑ being inconsistent, Lewis concludes that ϕ_M must be false if it is to belong to ϑ. Lewis further makes the point that to argue for ϕ_M's truth in ϑ would be to "…assume what is to be proved…of ϕ_M" (1979, p. 376). ϕ_M makes a statement about consistency (truth), but this does not make it true. The *reductio* proof must then fail yet again. Since it fails for both mentioned cases, on behalf of differentiating machines and their Gödel formulae, the argument fails and CTM stands yet to be refuted. (Lewis 1979, p. 376)

3.4.1 Lucas' rejoinder to Lewis

Lucas submitted a response to Lewis' new objection (Lucas 1984). He argues that the first case is erroneous since Lewis fails to adhere to the dialectic form of the argument. The machine he is up against is not O_M, rather it is O_M^M (since it is O_M plus ϕ_M, it is O_M^M). We are then forced to the second alternative. Lewis believes ϕ_M must be false if it is to be in O_M^M (Gödel-2), which in effect is the belief that the human mind has no warrant for assuming its own consistency (that is essentially what it is doing when it contains ϕ_M), even *if* it in fact is consistent. We have already seen Lucas argue for our permission to claim consistency. In this paper, Lucas responds by saying that if O_L^M is inconsistent there must be a proof of falsity for ϕ_M in O_L, making O_L inconsistent. On Lucas' assumption that O_L is consistent, the argument must again lead to a contradiction since it leads to O_L being inconsistent. (Lucas 1984, p. 191)

3.4.2 Lucas and Lewis: discussion

Lewis argues from the point of view that even *if* the mind is consistent, the argument that Lucas is not a machine fails since he cannot consistently claim that he is. Lucas' response concerning the two alternatives is valid in one respect. In the first case Lewis in fact fails to comply with Lucas' dialectical argument. The computationalist is allowed to submit *one* machine for Lucas to examine. If they submit O_M plus ϕ_M, it is in fact O_M^M that is submitted, in line with (iii). However, in the second case it truly boils down to whether we are allowed to consistently assert our consistency or not (already assuming that we are consistent). Lewis believes ϕ_M must be false in O_M^M (=O_L^M in the argument), which Lucas finds implausible. Lucas believes it is completely fine for O_L^M to contain ϕ_M, the formula stating O_L's truth. The debate is once again turned into a debate about human consistency and whether we can claim it or not. The second alternative succeeds, and therefore also Lewis' objection. It succeeds because Lucas must assume that he contains a *true* statement (it is true from the first premise) that states his consistency. This assumption cannot be hold conclusively; it lacks any proof. It is seen once again that the argument fails without this assumption. Lucas insists that there must exist a proof for the falsity of ϕ_M in O_L for the argument to fail. That is false in this argument with presumed premises. We already know that ϕ_M is true in O_L (this must be the case because of (i), it is

assumed that Lucas is sound), which is why we cannot accept the argument. If Lucas is sound, he cannot hold a true statement that he is (Gödel-2). Lucas has confused whom has the burden of truth, and premise (i) with reality. For us to conclude that ϕ_M is in fact true in O_L we must *know* that we are consistent, not merely assume. We do not know this, and it is a proof for *truth* in O_L that is needed for Lucas' argument to succeed. This proof is missing, and therefore the argument is inconclusive.

3.5 Concluding comments on the Lucas-Penrose argument

3.5.1 Reiteration of the Lucas-Penrose argument and its implications

Before giving a conclusion of the Lucas-Penrose argument, a short reiteration of the argument might be helpful: Lucas and Penrose claims that a machine capturing our reasoning is consistent, and that since a formal system cannot decide its Gödel formula (Gödel-1), which Lucas and Penrose claims we can see the truth of (and hence, decide), a consistent formal system (or sound computation) *cannot* capture our reasoning. (Lucas 1961) (Penrose 1994, §2.5)

In claiming that the formal system capturing our reasoning is consistent it is implied that we are formally consistent. It is also implied that we can prove the consistency of a system or computation to know that they have an undecidable formula in the first place. Since if we are machines we cannot know which machines we are, we *must* prove the consistency of a system without relying on the assumption of our consistency. Further, in claiming that we can see the truth of a Gödel formula from outside the system, it is also implied that we are allowed informal arguments when the machine is only allowed formal ones. Our notion of "truth" of the Gödel formula is based on Gödel-1's implications and the knowledge of a system's consistency. The machine's incapacity to decide (or "see the truth") of its Gödel formula is a formal result of Gödel-1; it is Gödel-1 in practice.

3.5.2 The Lucas-Penrose argument: conclusion

There are many objections to the Lucas-Penrose argument, and the ones *I* deem the most important have now been discussed. The main problems concern consistency – we must *prove* that the putative system is consistent, *and* we must assume that we are. If we cannot show that *F* or *A* are consistent, the argument fails. Because of Gödel-2, we cannot formally prove that a system is consistent. And if we in fact are computational systems, we cannot deduce which system we are. This contributes to us not *knowing* that *F* or *A* are consistent (they are corresponding to us, and we cannot assume what is to be proven). *Any* doubt of as to our consistency is enough for the argument to fail because we must *know* (unassailably believe) that we are consistent. These are the two main problems. If we require formal proofs to these claims, the Lucas-Penrose argument fails. Even Lucas himself acknowledges the problem of proving a system's consistency, though it is assumed by Lucas that we are *not* the putative system (because if we were, he would claim that we could know it is consistent):

> "Thus in order to fault the machine by producing a formula of which we can say both that it is true and that the machine cannot produce it as true, we have to be able to say that the machine (or, rather, its corresponding formal system) is consistent; and there is no absolute proof of this." (Lucas 1961, p. 10).

We may conclude that we cannot formally prove the consistency of a consistent formal system. Even if we allow Lucas the knowledge of whether the examined machine can decide its formula or not, it is still required to assume that we are consistent (Lucas also implies this in his response to Benacerraf (Lucas 1968, p. 157–158)). I will now discuss whether we should be allowed informal arguments for these claims or not.

Lucas argues that we are consistent because we contain a certain self-referential ability that is a product of our self-consciousness, and that this ability lets us see the truth of the self-referential Gödel formula. It is also argued by Lucas that our consistency is related to our natural *strive* to be consistent. We are not *knowingly* inconsistent, and we remove any observed inconsistencies (Lucas 1961, p. 10–11, 14–15). An argument against this is that unknowingly holding two contradicting

mathematical statements as true *is* an inconsistency, even if it is unknowingly. A human may hold a *generally* consistent framework of beliefs and still hold two contradictory beliefs, which in effect makes him inconsistent (if we are judged on the same formal basis as machines). The principle of explosion might not be as "generally" applicable to humans, but it is well agreed upon that it is usual for humans to hold various inconsistent beliefs at once. A once inconsistent system *is* an inconsistent system. Therefore, if we are judged on the same basis as machines, we are inconsistent. Whether or not we *should* be judged on the same basis as machines, I cannot provide an answer. Nevertheless, it is still an argument against human consistency.

We use informal arguments all the time, and I am sure we often arrive at *truth* with them. They are an important part of our everyday reasoning, and I do not believe the mentioned critics deny this. The problem lies in deriving formal conclusions from informal premises. It is normal to use informal explanations for formal proofs, such as I have done with Gödel's theorems in this paper, but it is not always the case for the contrary. Lucas' arguments are not formalizable; they *rely* on informality. The case for formal consistency claimed by informal proofs is problematic, just as Lucas and Penrose's sense of "seeing the truth" of a Gödel formula in a formal system, when the system is only allowed formal proofs. If there is a lack of a cogent argument for *knowing* that we are consistent, the Lucas-Penrose argument does not refute CTM. This is if we accept that Lucas and Penrose in fact could prove a formal system's consistency. The burden is on Lucas and Penrose to show that we are consistent, not for computationalists to prove otherwise.

In denying the arguments that we *know* that humans are consistent *and* the assumption that we can prove so, I also deny the legitimacy of the Lucas-Penrose argument. It does *not* refute CTM. I am conscious, but I do not know that a machine could not be.

4. The new Penrose argument

4.1 The Penrose-argument

I have just showed how the Lucas-Penrose argument must assume that we are consistent, and that we have proof for a putative system's consistency. The second claim is stronger – there are no formal proofs for such consistency. Penrose is aware of this, and therefore decides to improve the Lucas-Penrose argument to make it independent of this requirement (Penrose 1994). To do this he sets up a new scenario in which it is not important whether we have any proof for consistency, efficaciously removing the requirement of the stronger premise. The following is the new Penrose-argument.

We construct a robot $\mathcal{R}$ with the set of mechanisms $\boldsymbol{M}$ as its basis for computation. The content of $\boldsymbol{M}$ corresponds to the (in principle available) human mathematical reasoning. We then present $\mathcal{R}$ with $\boldsymbol{M}$, and the premise $\mathcal{M}$ which states that "$\boldsymbol{M}$ is $\mathcal{R}$*'s* underlying mechanisms". (We will soon see that whether in fact $\mathcal{R}$ was constructed according to $\boldsymbol{M}$ and whether $\mathcal{R}$ believes it to be true or not is of no importance, which is the main strength of the argument.) Assertions that $\mathcal{R}$ deem unassailable are denoted by the symbol $*$. $\mathbb{F}(\boldsymbol{M})$ is the formal system based on $\boldsymbol{M}$. The theorems of $\mathbb{F}(\boldsymbol{M})$ are the (1) unassailable $*$-assertions derivable from $\boldsymbol{M}$ and the (2) propositions obtainable from the $*$-assertions by way of elementary logic. $\mathcal{R}$'s unassailably deduced assertions are called $*$-assertions. Assertions $\mathcal{R}$ deem unassailably deductible from the assumption of $\mathcal{M}$ are then called $*_{\mathcal{M}}$-assertions, i.e., the assertions $\mathcal{R}$ believes to be *unassailably* true *if* the premise $\mathcal{M}$ is assumed to be true.

Believing in its own soundness *and* assuming $\mathcal{M}$, $\mathcal{R}$ believes in the soundness of the formal system $\mathbb{F}(\boldsymbol{M})$. Hence, $\mathcal{R}$ also believe in the Gödel formula $G_{(\mathbb{F}(\boldsymbol{M})}$ that is implied by $\mathbb{F}(\boldsymbol{M})$'s soundness. The assumption of $\mathcal{M}$ is necessary for $\mathcal{R}$ to have the assertion of $G_{(\mathbb{F}(\boldsymbol{M})}$, without it a contradiction emerge: if $\mathcal{R}$ were to believe in its own soundness and that $\mathbb{F}(\boldsymbol{M})$ is its underlying system, it would subsequently believe in the Gödel formula $G_{(\mathbb{F}(\boldsymbol{M})}$ all the while forced to believe that $G_{(\mathbb{F}(\boldsymbol{M})}$ lies outside of its belief system. Hence, the machine does not believe in $\boldsymbol{M}$ as its underlying rules (Penrose 1994, p. 163). $\mathcal{R}$'s belief in $G_{(\mathbb{F}(\boldsymbol{M})}$ rests on the assumption of $\mathcal{M}$.

We now construct a new formal system $\mathbb{F}_{\mathcal{M}}(\boldsymbol{M})$. $\mathbb{F}_{\mathcal{M}}(\boldsymbol{M})$ is equivalent to $\mathbb{F}(\boldsymbol{M})$, except for the $*_{\mathcal{M}}$-assertions replacing the $*$-assertions (viz., in the theorems). Assuming $\mathcal{M}$, $\mathcal{R}$ should see no problems with the system $\mathbb{F}_{\mathcal{M}}(\boldsymbol{M})$ representing its $*_{\mathcal{M}}$-assertions.

Accepting $\mathbb{F}_{\mathcal{M}}(\boldsymbol{M})$ as sound on the assumption of $\mathcal{M}$ and belief in its own soundness, $\mathcal{R}$ subsequently accepts the Gödel formula $G_{\mathbb{F}_{\mathcal{M}}(\boldsymbol{M})}$ as an $*_{\mathcal{M}}$-assertion (and therefore as true) as it is a consequence of the system's soundness. This gets Penrose the sought for contradiction. $G_{\mathbb{F}_{\mathcal{M}}(\boldsymbol{M})}$ cannot be true if $\mathbb{F}_{\mathcal{M}}(\boldsymbol{M})$ is sound (according to Gödel-1), but $G_{\mathbb{F}_{\mathcal{M}}(\boldsymbol{M})}$ must be true if it is a consequence of $\mathcal{M}$ and $\mathcal{R}$ is sound.[17] The only option for $\mathcal{R}$ is then to reject the hypothesis $\mathcal{M}$ – it cannot have been constructed with $\boldsymbol{M}$. Exploring the logical consequences of the *assumption* of $\mathcal{M}$ is enough for the contradiction to rise. It does not matter whether $\boldsymbol{M}$ are the system's underlying mechanisms or not, it must reject the possibility. Rejecting $\mathcal{M}$, $\mathcal{R}$ rejects the premise "$\boldsymbol{M}$ is $\mathcal{R}$*'s* underlying mechanisms". Conclusion – no *knowably* sound computational system can encapsulate the (in principle available) human mathematical reasoning. (Penrose 1994, §3.1–§3.16, §3.23)

In *Beyond the Doubting of a Shadow* (1996) Penrose formulates his intended Penrose-argument in one paragraph (A):

(A) "Though I don't know that I necessarily am F, I conclude that if I were, then the system F would have to be sound and, more to the point, F' would have to be sound, where F' is F supplemented by the further assertion "I am F". I perceive that it follows from the assumption that I am F that the Gödel statement G(F') would have to be true and, furthermore, that it would not be a consequence of F'. But I have just perceived that "if I happened to be F, then G(F') would have to be true", and perceptions of this nature would be precisely what F' is supposed to achieve. Since I am therefore capable of perceiving something beyond the powers of F', I deduce that, I cannot be F after all. Moreover, this applies to any other (Gödelizable) system, in place of F." (1996, 3.2)

4.1.1 What the Penrose-argument escapes

The argument avoids the problem of having to prove the soundness (hence, consistency) of a system. It is now enough for us to *know* (unassailably believe) we are sound, since it is then implied that $\boldsymbol{M}$ (and $\mathbb{F}(\boldsymbol{M})$ and *A*) is sound as well. Lucas must no longer claim that we are able to determine a system's consistency from outside with informal arguments. The burden of $\mathbb{F}$ needing to prove that its corresponding system is sound is no longer there. One of the main objections concerning consistency and informality is removed.

It is still *assumed* that the set of human reasoning mechanisms $\boldsymbol{M}$ is sound, but $\mathbb{F}$ is not forced to prove it any longer. It is enough for $\mathbb{F}$ to *believe* it is sound and assume $\mathcal{M}$. If a robot is presented with a sound system's mechanisms that are equivalent to its own mechanisms it would have to reject them (1994, §3.2, p. 163 & 165). This problem is also escaped by the mere assumption of $\mathcal{M}$.

4.1.2 The Penrose-argument formalized

Let *F* be a formal system. S(x) is a formalization of "x is arithmetically sound" and T(x) is a formalization of "x is true". $\boldsymbol{M}$ is the (in principle available) human mathematical reasoning. $F_{\boldsymbol{M}}$ means that *F* is based on $\boldsymbol{M}$ (practically $F = \boldsymbol{M}$). $F^{+}_{\boldsymbol{M}}$ is the system $F_{\boldsymbol{M}}$ supplemented with $S(F_{\boldsymbol{M}})$.[18] $G_{F+\boldsymbol{M}}$ is F^{+}'s Gödel formula. It is assumed that $F_{\boldsymbol{M}}$ (and therefore $F^{+}_{\boldsymbol{M}}$) is capable of PA. The Penrose-argument may be formalized in the following way, see Table 2:

Table 2: Penrose-argument formalized.[19]

$S(\boldsymbol{M})$	Premise ($\boldsymbol{M}$ is sound)	(1)
$(S(\boldsymbol{M}) \,\&\, F_{\boldsymbol{M}}) \rightarrow S(F_{\boldsymbol{M}})$	Premise (If $\boldsymbol{M}$ is sound and *F* is based on $\boldsymbol{M}$, $F_{\boldsymbol{M}}$ is sound)	(2)
$S(F_{\boldsymbol{M}}) \rightarrow S(F^{+}_{\boldsymbol{M}})$	Premise (If $F_{\boldsymbol{M}}$ is sound, $F^{+}_{\boldsymbol{M}}$ is sound)	(3)

[17] As it is a consequence of $\mathbb{F}_{\mathcal{M}}(\boldsymbol{M})$, $\mathcal{R}$ should deem it an $*_{\mathcal{M}}$-assertion, and hence a theorem of $\mathbb{F}_{\mathcal{M}}(\boldsymbol{M})$. However, no sound formal system can have its own Gödel formula as a theorem.

[18] $(F_{\boldsymbol{M}} + S(F_{\boldsymbol{M}})) = F^{+}_{\boldsymbol{M}}$.

[19] As with table 1: this is my own formalization, and other formalizations are also possible.

$S(F^+_{\mathcal{M}}) \rightarrow T(G_{F+\mathcal{M}})$	Premise (Because $F^+_{\mathcal{M}}$ is sound, $G_{F+\mathcal{M}}$)	(4)
$S(F^+_{\mathcal{M}}) \rightarrow F^+_{\mathcal{M}} \nvdash G_{F+\mathcal{M}}$	Premise (If $F^+_{\mathcal{M}}$ is sound, $F^+_{\mathcal{M}}$ cannot prove $G_{F+\mathcal{M}}$ (Gödel-1))	(5)
$F^+_{\mathcal{M}} \nvdash G_{F+\mathcal{M}} \rightarrow \sim T(G_{F+\mathcal{M}})$	Premise (If $F^+_{\mathcal{M}}$ cannot prove $G_{F+\mathcal{M}}$, $G_{F+\mathcal{M}}$ is not true)	(6)
$\sim(S(\mathcal{M}) \,\&\, F_{\mathcal{M}})$	Premise (We see that these two together leads to a contradiction)	(7)

Premise 1: we know that our reasoning is sound.
Premise 2: we know that if F is based on our reasoning ($F_{\mathcal{M}}$, practically $F = \mathcal{M}$), F is also sound.
Premise 3: if $F_{\mathcal{M}}$ is sound the extension $F^+_{\mathcal{M}}$ is sound, as it is $F_{\mathcal{M}}$ plus the true statement that $F_{\mathcal{M}}$ is sound. "Supplementing a sound system with a true statement yields a sound system." (Chalmers 1995, 3.2).
Premise 4: since we are (practically) $F^+_{\mathcal{M}}$ we know that $G_{F+\mathcal{M}}$ must be true since it follows from the assumption of $F_{\mathcal{M}}$, and we know that $F_{\mathcal{M}}$ is sound (since $F = \mathcal{M}$ and $\mathcal{M}$ corresponds to and idealized human).
Premise 5 & 6: since we know that $F^+_{\mathcal{M}}$ is a sound formal system it cannot contain (as a theorem) its Gödel formula according to Gödel-1. Hence, $G_{F+\mathcal{M}}$ cannot be true. This leads us to a contradiction.
Conclusion: we know that $G_{F+\mathcal{M}}$ is true since it follows from us being sound, but (6) states that it is not true. It must follow that F cannot contain us ($F \neq \mathcal{M}$) since our reasoning is sound and we can see the truth of $G_{F+\mathcal{M}}$, unlike $F_{\mathcal{M}}$ that is bound by Gödel-1. We conclude that $\sim F_{\mathcal{M}}$. F is *any* formal system, and therefore *no* formal system can contain our reasoning. (Penrose 1994, §3.23)

4.2 Chalmers' objection

Chalmers' objection concerns the unassailable belief in human consistency. He argues that a belief system believing in its own consistency leads to a contradiction. I will now give the argument. (Chalmers 1995, 3.6– 3.14)

Let B stand for "Belief". B(n) is the belief in the statement with the Gödel number n. "$\vdash$n" means that the system can unassailably assert n. The discussed belief system is capable of PA. We have the following assumptions of the system:

$\vdash A \rightarrow \vdash B(A)$	(if the system can assert A, it can assert its belief in A)	(1)
$(\vdash B(A_1) \,\&\, B(A_1 \rightarrow A_2)) \rightarrow B(A_2)$	(the system is capable of modus ponens)	(2)
$\vdash B(A) \rightarrow B(B(A))$	(the system believes in its own beliefs; it is aware of (1))	(3)
$\vdash \sim B(\text{false})$	(the system asserts that it is not inconsistent)	(4)

With the three first assumptions of the system, we add the fourth assumption, the system's belief in its own consistency. Chalmers claims a contradiction is derivable from (4). More literally, (4) says "I unassailably assert that I do not believe in a contradiction". (Chalmers 1995, 3.8–3.9)

We derive the contradiction by adding a fifth premise, the G-formula.

$\vdash G \rightarrow \sim B(G)$	(the G-formula)	(5)

(5) states "I do not believe G". G is the Gödel number for the Gödel formula in the system. The self-referential formula is a product of a diagonalization, see Chalmers 3.10 (1995) for more information on this. The system knows about Gödel-1, and therefore rejects its belief in G to avoid inconsistency. However, since the system believes it is sound, it must believe in containing G and that G is true in the system. The system must reject its belief in G on pain of inconsistency, and at the same time it must believe in (the truth of) G if it believes it is sound – two contradictory beliefs. The system's belief in its own soundness leads to a contradiction.[20] Chalmers' formalized proof of the contradiction derivable from (4) continues:

$\vdash B(G) \rightarrow B(\sim B(G))$	(from premises (1), (2) and (5))	(6)
$\vdash B(G) \rightarrow B(B(G))$	(from premise (3))	(7)
$\vdash B(G) \rightarrow B(\text{false})$	(from premises (2), (6) and (7))	(8)

[20] Penrose himself has made an argument with this sort of reasoning (1994 §3.2, p. 163 & 165) (see 4.1).

⊢B(false) → B(G)	(from premise (2) and ⊢B(false → G))	(9)
⊢G → ~B(false)	(from premises (5), (8) and (9))	(10)
⊢B(G)	(from premises (1), (4) and (10))	(11)
⊢B(false)	(from premises (9) and (12))	(12)

Conclusion: it is contradictory for a belief system to believe in its own consistency. (Chalmers 1995, 3.10–3.11)

4.2.1 Penrose's response

Penrose responds by clarifying which belief-formula he is considering that the system asserts in his arguments. It is strictly "P-sentences" (Π_1-formulae) the system is asserting. Penrose (informally) defines a Π_1-formula as determining whether a certain computation of a Turing machine halts or not (*A* is a Π_1-formula). Then, "all its [the pertinent belief system] *outputs* must be assertions as to the validity of particular P-sentences" (Penrose 1996, 3.8). The $*$-assertions and $*_{\mathcal{M}}$-assertions are all Π_1-formulae, and the "humanly available mathematical reasoning" $\boldsymbol{M}$ used as Penrose's systems underlying rules is *only* used in terms of deciding Π_1-formulae. (Penrose 1996, 3.6–3.8)

In restricting the belief system to exclusively contain Π_1-formulae, Penrose argues that the diagonalization used for creating the G-formula is no longer viable for usage in the proof, as the G-formula itself is not a Π_1-formula. As it is not, the G-formula cannot be part of the systems Penrose is considering in §2.5 and §3.23 in *SOTM*. If the G-formula that Chalmers uses is non-applicable in Penrose's argument, Chalmers' argument fails. Conclusion: restricting the system to only Π_1-formulae avoids Chalmers' contradiction. (Penrose 1996, 3.9–3.13)

4.2.2 Lindström's response

Logician Per Lindström responds to Chalmers' objection made against the Penrose-argument. Lindström argues that Chalmers' G-formula might not be well-defined (unambiguous). "B(G)" does not have a well-defined meaning, as there exists G's (in the system's language) that are missing a well-defined truth value in B(G). If G in B(G) contains the symbol B, the meaning of the formula becomes confused. The G-formula then says, "I do not believe that this formula [G, referring to itself] is true". It is unclear what that means. When there exists a G such that it satisfies the G-formula, the system is led to a contradiction, in line with Chalmers' proof. However, the formal contradiction derived from (4) cannot be blamed on the system's "belief" in its consistency, because the G-formula in that interpretation makes no sense. The contradiction is derived because G satisfies the G-formula, at the cost of making no semantical sense. (Lindström, 2001, p. 428)

A diagonalization that is of similar structure but has a well-defined truth value for every G ("F-Pr" stands for "provable in F") is:

(5') F ⊢ G ↔ ~F-Pr(G).

For every G it is clear whether it is provable in F or not. This cannot be said for every G in (5). It is essentially "Provable-in" versus "Belief-in". The *intention* of Chalmers' proof ultimately fails, although the formal proof leads to inconsistency. (Lindström, 2001, p. 248–249)

4.2.3 Penrose, Chalmers and Lindström: discussion

One of the main strengths of the argument is that it is not only applicable to a computational system of beliefs (assertions), but to *any* belief system. It attacks the main problem of the new Penrose argument, instead of possible inconsistencies or technical problems in Penrose's text, or other problems not concerning the actual *idea* of the argument. Chalmers writes "…I have come to believe that the greatest vulnerability in this argument lies in the assumption that we know (unassailably) that we are consistent." (Chalmers 1995, 3.6). I agree with Chalmers; it is a very strong assumption for which there exists no verification for other than informal reasoning. Now to the discussion of the argument's content.

Penrose's response succeeds since it is true that Chalmers' G-formula is not a Π_1-formula (which we are restricted to) and is therefore not functioning as a basis for contradiction in the system, since the system cannot contain it. However, the restriction renders the Penrose-argument idle. This is due to the very limits Penrose himself forces on the system. In limiting the pertinent system ($\mathbb{F}(\boldsymbol{M})$ and $\mathbb{F}_{\mathcal{M}}(\boldsymbol{M})$) to Π_1-formulae, we must also limit the idealized human's mechanisms $\boldsymbol{M}$ to Π_1-formulae. In doing this, we ($\boldsymbol{M}$) would have to transcend this limit to arrive at the conclusion of $\mathbb{F}$ not containing $\boldsymbol{M}$. The required diagonalization for seeing that $\mathbb{F}$ must reject $\boldsymbol{M}$ will not end up in a Π_1-formula (the only type of formula we are allowed to assert), and hence cannot be used in our reasoning for $\mathbb{F}$ rejecting $\boldsymbol{M}$. Penrose's argument works against himself. If we accept Penrose's limitation, Chalmers' argument is refuted and the Penrose-argument fails.

Concerning the legitimacy of Chalmers' proof. As Lindström's objection to Chalmers' G-formula stands, we might still be missing a *formal* proof against the assumption of human consistency. This might be the case, but the burden of proof for unassailably believing in human consistency remains for the Penrose-argument to be of any impact on CTM. Although Chalmers' intention of his formal proof fails (that a system's *belief in its own consistency* leads to a contradiction), there might yet exist other formal objections to the Penrose-argument. We will now look at two such objections made by Lindström, Bringsjord and Xiao.

4.3 Lindström's objection

The Penrose-argument relies on (3) in Table 2: $S(F_{\boldsymbol{M}}) \rightarrow S(F^{+}_{\boldsymbol{M}})$. By utilizing the implications of Gödel-2, Lindström argues that this premise is not provable. Lindström shows how regardless of Penrose's definition of soundness, (3) is not valid. Subsequently, since (3) is required for the proof, the proof is invalid as well. (Lindström 2001, p. 245–247)

The Penrose-argument is only meant to concern "soundness" in terms of *arithmetic soundness*[21], therefore, only the counterargument to (3) concerning arithmetic soundness will be detailed here. Instead of (3), we may use the weaker premise (3') ("Con" standing for "consistent"):

$$S(F_{\boldsymbol{M}}) \rightarrow \mathrm{Con}(F^{+}_{\boldsymbol{M}}) \qquad (3')$$

If (3') fails, (3) fails.[22] From Gödel-2 it follows that if F is consistent (and capable of PA), then F+~Con(F) is a valid extension of F. However, F+~Con(F)+Con(F+~Con(F))[23] is *not* consistent, even if it is derived from the consistency of F. Lindström comments that "The (straightforward) counterexamples F to (1) [(3') here] are of the form E + ¬Sd(E)[¬ = ~]; they are sound, in the sense that Sd(E + ¬Sd(E)) is true, but obviously not true." (Lindström 2006, p. 232). Thus, it does not follow from the soundness of $F_{\boldsymbol{M}}$ that $F^{+}_{\boldsymbol{M}}$ is consistent, and certainly not that it is sound. When (3) fails, so does the Penrose-argument. (Lindström 2001, p. 246)

Franzén explains this type of argument well - "That S is consistent does not, as we know from the second incompleteness theorem itself, rule out that S proves false theorems" (1995, p. 106).

4.4 Bringsjord and Xiao's objection

Selmer Bringsjord and Hong Xiao formulated another technical objection against Penrose's new argument. They argue that the Penrose-argument is not a contradiction at all. The contradiction is

[21] Or "Π_1-sound": every provable Π_1 formula of the system is true (Lindström 2005, p. 245–246). See 2.1 for notes on arithmetic soundness.

[22] If F^{+}*'s* consistency does not follow from the soundness of F, neither does soundness.

[23] Essentially $F^{+}_{\boldsymbol{M}}$, but referring to consistency instead of soundness. It should also be noted that this is a Π_1-formula. (Franzén 2005, p. 21)

derived from an ambiguous use of *truth*.[24] A hypothetical mathematician looking at the argument can conclude that:

(1) "T($G_{F+\mathcal{M}}$) is true", and
(2) "$F^{+}_{\mathcal{M}} \nvdash G_{F+\mathcal{M}}$ and $F^{+}_{\mathcal{M}} \nvdash \sim G_{F+\mathcal{M}}$" ($G_{F+\mathcal{M}}$ is undecidable in $F^{+}_{\mathcal{M}}$).

But (1) is *de facto* a claim about *satisfaction*. The claim is essentially (where *I* is an interpretation)

(1') "$I \vDash G_{F+\mathcal{M}}$ is true" (there exists a model which satisfies the Gödel formula),

which is a *meta*-mathematical assertion, not the (non-meta) logical type of assertion that is being implied by Penrose. It goes beyond what the proof allows for. Even if (2) holds, it does not follow that the formal system cannot prove (1'), which is the sense of truth Penrose is in fact using. "Penrose conflates proofs within a fixed system with meta-proofs" (Bringsjord & Xiao 2000, p. 325). There is no contradiction is Penrose's argument. (Bringsjord & Xiao 2000, p. 324–325)

4.5 The Penrose-argument: conclusion

The Penrose-argument is a great improvement on the Lucas-Penrose argument, as the problem of having to *prove* a system's consistency is gone. However, Lindström, Bringsjord and Xiao's objections showed that the argument is nevertheless invalid. Lindström showed that one of the premises is not provable, and Bringsjord and Xiao showed how Penrose conflates meta-mathematical assertions with "standard" logical assertions. Even if we chose to ignore these technical issues, it is still required of us to *know* (or "unassailably believe") that we are sound. Any hesitancy as to the assumption's legitimacy is enough for the Penrose-argument to fail. Since no evident proof for the claim has been provided, we must conclude that the Penrose-argument at best is inconclusive. It is not necessary to have a proof against Penrose's needed assumption for the argument to be futile, it is enough for the premise to lack any evidence itself. The failure of Chalmers' counterargument is then of no worries to CTM, because Penrose still must make the unwarranted assumption that we are sound.

I made an argument against the assumption of human consistency in 3.5.1. Bringsjord and Xiao provide one more: there might be hidden contradictions in our mathematical understanding that are a cause for inconsistency. There are in fact such contradictions that are yet to be solved, such as Yablo's paradox. (Bringsjord & Xiao 2000, p. 322–324)

In lack of convincing proof for an unassailable belief in human consistency, the Penrose-argument stands yet to refute CTM.

5. Summary and Conclusion

The Gödel-arguments made by Penrose and Lucas are fails because they require the assumption of unjustified premises about human consistency. The Lucas-Penrose argument is inconclusive because it must assume that humans are consistent *and* that we can prove the consistency of a formal system corresponding to us (without knowing that it corresponds to us). The Penrose-argument is inconclusive because it must assume that humans are sound (hence, consistent).

We have explored various objections against the Lucas-Penrose argument made by Benacerraf and Lewis and Lucas himself. The strongest objection is that we cannot prove that a system is consistent. Lucas invokes the dialectics of his argument, which may be argued to free him from such a burden. However, it is still required of him to *know* that we are consistent. There are no formal proofs to that claim, and the informal proofs are not adequate for supposing such a strong assertion. Therefore, the Lucas-Penrose does not disprove CTM.

We then inquired into the Penrose-argument and its objections. The Penrose-argument does not

[24] This was discussed earlier in the objections to the Lucas-Penrose argument. This objection pinpoints what is in fact the difference in the "truth"-claims.

require one to prove the putative system's consistency, it is enough to know that *we* are sound. An argument was made by Chalmers to try and prove that this leads to a contradiction, but it was refuted by Lindström. We then saw two objections by Lindström, Bringsjord and Xiao as to the validity of the Penrose-argument. It was concluded that the argument failed. One of the premises is unprovable and one is a *meta*-assertion mixed in with the "non-meta" assertions, showing how the proof in fact does not lead to any inconsistencies. It was also concluded that even if we neglect these problems of the Penrose-argument, it still presupposes that the human mind is sound, which is yet to be proven. The Lucas-Penrose argument therefore fails as well.

The Gödel-arguments submitted by Lucas and Penrose are at most enthymemes, as they rely on implied premises concerning consistency. As no convincing validations of said premises are provided, they do not refute CTM. This does not necessarily mean that there exist no other forms of Gödel-arguments that can refute CTM. Other professionals, such as Bringsjord, also believe CTM can be refuted with Gödel-arguments (Bringsjord & Xiao 2000, p. 326). As it stands, no such Gödel-arguments have been made, and the Computational Theory of Mind is still up for debate.

References

Alexander S. A. (2013). A Machine That Knows Its Own Code. *Studia Logica*, 102 (3), 567–576

Barker-Plummer D., Barwise J., Etchemendy J. (2009). *Language, Proof and Logic* (2nd ed.). Center for the Study of Language and Information

Benacerraf P. (1967). God, the Devil, and Gödel. *The Monist*, 51(1), 9–32

Bringsjord S. & Xiao H. (2000). A Refutation of Penrose's Gödelian Case Against Artificial Intelligence. *Journal of Experimental and Theoretical Artificial Intelligence*, 12, 307–329

Britannica, T. Editors of Encyclopaedia (2012, January 6). Formal system. *Encyclopedia Britannica*

Britannica, T. Editors of Encyclopaedia (2017, June 12). Reductio ad absurdum. *Encyclopedia Britannica*

Boyer D. (1983). J. R. Lucas, Kurt Godel, and Fred Astaire*. Philosophical Quarterly. 33, 147–159*

Chalmers D. J. (1995). Minds, Machines, And Mathematics. *PSYCHE*, 2 (9)

Coder D. (1969). Gödel's Theorem and Mechanism. *Philosophy*. 44, 234–237

De Mol. L. (Winter 2019 Edition). Turing Machines. *The Stanford Encyclopedia of Philosophy*, Edward N. Zalta (ed.)

Feferman S. (1995). Penrose's Gödelian Argument, *PSYCHE*, 2(7)

Franzén T. (2005). *Gödel's Theorem: An Incomplete Guide to Its Use and Abuse*, Massachusetts: A K Peters, Ltd.

Hosch, W. L. (2010, December 1). Peano axioms. *Encyclopedia Britannica*.

LaForte G., Hayes J. P., Ford K. M. (1998). Why Gödel's theorem cannot refute computationalism. *Artificial Intelligence*, 104(1–2), 265–286

Lewis D. (1969). Lucas against Mechanism. *Philosophy*, 44(169), 231–233

Lewis D. (1979). Lucas against Mechanism II. *Canadian Journal of Philosophy*, 9(3), 373–376

Lindström P. (2001). Penrose's new argument. *Journal of Philosophical Logic*, 30(3), 241–250

Lindström P. (2006). Remarks on Penrose's "New Argument". *Journal of Philosophical Logic*, 35, 231–237

Lucas J. R. (1961). Minds, Machines and Gödel. *Philosophy*, 36(137), 112–127

Lucas J. R. (1968). Satan Stultified: A Rejoinder to Paul Benacerraf. *The Monist*, 52(1), 145–158

Lucas J. R. (1970). Mechanism: A rejoinder. *Philosophy*, 45(172), 149–151

Lucas J. R. (1984). Lucas against Mechanism II: A Rejoinder. *Canadian Journal of Philosophy*, 14(2), 189–191

Penrose R. (1989). *The Emperor's New Mind*. Oxford: Oxford University press

Penrose R. (1994). *Shadows of the Mind*. London: Vintage

Penrose, R. (1996). Beyond the Doubting of a Shadow. *PSYCHE*, 2(23)

Putnam H. (1960), Minds and Machines. *Dimensions of Minds*, 138–164

Raatikainen P. (Spring 2021 Edition). Gödel's Incompleteness Theorems. *The Stanford Encyclopedia of Philosophy*, Edward N. Zalta (ed.)

Rescorla M. (Fall 2020 Edition). The Computational Theory of Mind. *The Stanford Encyclopedia of Philosophy*, Edward N. Zalta (ed.)

Shapiro S., Teresa K. K. (Spring 2021 Edition). Classical Logic. *The Stanford Encyclopedia of Philosophy*, Edward N. Zalta (ed.)

W. K. Macura. (n.d.) Abstract Machine. From *MathWorld*--A Wolfram Web Resource

Wang H. & Schagrin M. L. (2011, January 27). *Metalogic*. *Encyclopedia Britannica*

Wilfrid H. (Winter 2020 Edition). Model Theory, *The Stanford Encyclopedia of Philosophy*, Edward N. Zalta (ed.)