The anti-mechanism argument based on Gödel's incompleteness theorems, indescribability of the concept of natural number and deviant encodings

Paula Quinon
Warsaw University of Technology
Department of Administration and Social Sciences
International Center for Formal Ontology

Abstract

This paper reassesses the criticism of *the Lucas-Penrose anti-mechanism argument based on Gödel's incompleteness theorems* formulated by Krajewski [17]. Krajewski claims that in order for *the argument* to work, *an extra-formal assumption* should be added: "the human mind is consistent". The assumption is "extra-formal", because it cannot be formalised and must be accepted on the basis of a non-formal insight. Thus – claims Krajewski – the Lucas-Penrose anti-mechanism argument, which relies on the formalisation of the whole process, fails to establish that human mind is not mechanistic.

An additional light on Krajewski's rejection of the anti-mechanism argument, is shed by *a corollary to this argument*: the human mind allegedly outperforms machines, because although there is no purely formal, exhaustive definition of natural numbers, human mathematicians can successfully work with natural numbers. Also in this case, as Krajewski claims, the anti-mechanism argument requires an additional assumption, such as "PA1 is complete" or "there exists a set of all natural numbers", which must be accepted without a full formal explanation of its meaning. Again, the anti-mechanism argument, which requires full formalisation, fails.

I agree with Krajewski that additional extra-formal assumptions are necessary to make the anti-mechanism argument, and its corollary, workable. However, I do not agree with Krajewski's statement that assumptions cannot be formalised. Instead, I propose an alternative problem with "extra-formal" assumptions that call into question the anti-mechanism argument, namely *circularity*. A machine cannot prove its own consistency; a human

mind, to show that is can outperform a machine in this respect, cannot simply bring in the assumption "I am consistent'. Such a reasoning is circular. Starting with an analysis of circularity, I propose a way of thinking about the interplay between informal and formal resources in mathematical reasoning.

Keywords: the Lucas-Penrose anti-mechanism argument, Gödel's incompleteness theorems, the concept of natural number, the concept of computation, Carnapian explications, conceptual engineering, conceptual fixed points, conceptual vicious circles, deviant encodings, structuralism, computational structuralism, the Church-Turing thesis

---

Contents

Introduction

The Lucas-Penrose anti-mechanism argument against computability of the human mind in a nutshell states what follows. According to Gödel's incompleteness theorems, there does not exist a (sufficiently rich) consistent theory that can prove its own consistency. However, mathematical practice shows that Gödel-type results are commonly proven by human mathematicians. In consequence, says the argument, human mathematicians are not describable as formal proof systems, nor are reducible to performing algorithms.

In [17], Krajewski criticies the Lucas-Penrose argument by claiming that Gödel's incompleteness theorems *stand alone* (as it is in the Lucas-Penrose case) are not sufficient for formulating the claim that the human mind is non-computational. The anti-mechanism argument based on Gödel's incompleteness theorems needs to be enriched by an extra-formal assumption. For instance, an assumption that the theory constituting the human mind is consistent.

In order to provide an additional context to his investigations, Krajewski [17], highlights the analogy between the claim that Gödel's incompleteness theorems imply the non-computational nature of the human mind, and the claim that "we [humans] cannot give a definition of the natural numbers as we understand them" [17, page 31]. The analogy goes as follows: in order to make a successful anti-mechanism argument based on Gödel's incompleteness theorems, one needs to assume – in addition to the formal counterpart – that the theory constituting the human mind is consistent. The fact that Gödel's argument can be iterated for increasingly rich theories is not sufficient for formulation of the anti-mechanism argument. The possibility to iterate increasingly rich theories, which all have a Gödel's sentence, and none of which proves its own consistency, is a formal process and as such can be executed with purely formal means. Thus, it does not say anything about computability or non-computability of the human mind. In order to be able to formulate the anti-mechanism argument, one needs to assume – for instance – that the human mind is consistent. Analogously, each definition of natural number ends up in a vicious circle of definitions, or – as Krajewski says – "our axioms [both the first-order ($PA1$) and the second-order Peano Arithmetic ($PA2$)] define numbers only when taken together with some background knowledge or apparatus that makes possible our intuitive grasp of numbers [such as the intuition that the first-order Peano's Arithmetic is complete or the intuition that there exists the set of all natural numbers being referred to in the background of the second-order Peano's Arithmetic]" [17, page 31]. In both cases, an immediate, but incorrect according to Krajewski, conclusion could be that "no computer can be taught our concept of a number" and that in consequence "we [humans] are better than any machine" [17, page 31].

In this paper, I observe that this analogy can be pushed further to a circular reasoning. In both cases, making an extra-formal assumption leads to a vicious circle because one assumes consistency of one's mind while proving that the human mind outperforms machines, or one assumes that the concept of set of natural numbers can be intuitively apprehended while defining

natural numbers. There exist studies showing that the method of conceptual analysis is particularly sensitive to falling into circular reasoning. The circularity related to the concept of natural number has been investigated in discussions about *computational structuralism* (Halbach & Horsten 2005 [14], Quinon & Zdanowski 2007 [36]). Computational structuralism is a position, according to which the concept of natural number and the concept of computation are closely related. More precisely, according to this position, an adequate account of natural numbers treats them as objects that can be used for computations. After a brief overview of the anti-mechanism argument and its criticism in **Section 1**, in **Section 2** I explain inter-relation and inter-definability between the concept of natural number and the concept of computation. In **Section 3**, I describe how the two concepts fall into a vicious circle of definition individually, and also while used in definition of one another.

Rescorla (2007 [41]) identifies problems with conceptual analysis related to the concept of computation, Quinon (2018 [38]) suggests that there is no fully satisfactory way out from vicious circles in definitions within conceptual analysis. Approaching the concept of computation and the concept of natural number from another methodological perspective, seems to be more fruitful. For instance, an interesting insight can be gained thanks to *conceptual engineering*. Both concepts have a form of what in the area of conceptual engineering is called "conceptual fixed point". A conceptual fixed point is an idea issued from conceptual engineering of moral concepts, where it is claimed that some basic moral concepts should not be engineered, but should always be understood in the most objective way (Eklund 2015 [10]). **Section 4** is devoted to presentation of the method of conceptual engineering and adequacy of conceptual fixed points for the concept of computation and the concept of natural number. As suggested by the phenomenon of conceptual fixed points, the only way out from vicious circles consists in an arbitrary decision what is the intended meaning of the given concept.

In **Section 5** I extend my methodological investigations into yet another method, and I discuss advantages of thinking about formalisation of the concept of computation in terms of Carnapian explications. It has been argued, for instance in (Quinon 2019 [39]), that a move from an intuitive concept of computation, used in everyday life, to a scientific or formal concept as stated by the Church-Turing thesis, follows the schema of a Carnapian explication. In **Section 6**, I extend the context of Carnapian explications of the temporary aspect. I realise that both, the concept of natural number

and the concept of computation, have been evolving in such a way, that their core meanings were shifting. I propose a hypothesis that at least a part of the confusion regarding the specificity of the conceptual structure of the concept of computation contributes to the confusion regarding the nature of human reasoning and the human mind. In consequence, I claim that – at least partially – the "feeling" that there exist non-computational processes comes from the complexity of the conceptual structure of the concept of computation.

In the final **Section 7**, I wrap up with the ways in which my observations regarding the concept of computation and the concept of natural number, could be used for understanding the reasons for which the anti-mechanism argument fails. I suggest a different reason from the one proposed by Krajewski, for which the extra-formal assumption prevents the anti-mechanism argument from success. Firstly, I claim that thanks to the method of Carnapian explications, it is highly possible to go from intuitive pre-scientific concept to a formal concept. Secondly, I observe that the extra-formal assumption after an arbitrary formalisation, leads to the vicious circle in reasoning. Therein lies the problem.

1. The Lucas-Penrose argument and its criticism

In this section, I present a brief overview of various versions of the anti-mechanism argument based on Gödel's incompleteness theorems, and the ways in which those arguments were criticised. In particular, I explicate Krajewski's way of refuting the argument. In my overview, I prioritise the authors to which Krajewski refers in his paper.

The first Gödel's incompleteness theorem says that in every sufficiently rich[1] consistent first-order theory [2] there exist statements that are true[3], but that cannot be proven within this theory. The second Gödel's incompleteness theorem says that every sufficiently rich consistent first-order theory cannot prove its own consistency.

Human mathematicians – states the anti-mechanism argument based on

[1] By a "sufficiently rich" one means that the formal system is able to express arithmetic of addition and multiplication.

[2] A formal system, or a theory, is a collection of axioms together with rules of inference. The importance of using first-order logic is because of the completeness of this logic.

[3] A statement is true, when it is satisfied in the intended model of the theory.

Gödel's incompleteness theorems – can fruitfully work with Gödel's incompleteness theorems, thus those mathematicians use a the resources from the outside the theory, for instance, they are able to refer to the intended model of arithmetic or recognize that the human mind is consistent. Thus, human mathematicians outperform machines, because – unlike machines – they are able to include in their reasoning such external resources.

The intuition that humans could prove theorems which machines could not has been present already in Turing (1950 [47])[4] and in Post (1941 [31])[5]. One of the most famous voices exploring the anti-mechanism argument based on Gödel's incompleteness theorems against the computational theory of mind – next to Hofstadter (1979 [15]) and Nagel and Newman (1958 [24], 1961 [25]) – Lucas (1961 [18], also 1968 [19], 1996 [21]) presented a "mathematical proof" of man's superiority over a machine. Lucas extended the applicability of Gödel's incompleteness theorems from formal systems to human subjects. On his view, humans are subjects to the same formal limits as machines. However, as Lucas observes, human mathematicians can prove Gödel's incompleteness theorem, which means, human mathematicians use extra-formal resources that enable them to perform such proofs.

Lucas' argument relies on the fact that Gödel's theorem(s) are formulated in purely formal terms. As Lucas observes himself, this is what differentiates the Gödel's results from the liar paradox. The liar paradox, stating that "This statement is untrue", is "viciously self-referential, and we do not know what the statement is, which is alleged to be untrue, until it has been made, and we cannot make it until we know what it is that is being alleged to be false" (Lucas 1990 [20, page 2]). Unlike the liar paradox, Gödel's theorem is formulated within a full-blooded system where it is clearly defined, which sentences are true and what does it mean to be provable. Lucas' claims that the fact that an (idealised) human mind, even if cannot prove the Gödel's theorem(s) for the given theory, can – thanks to its additional non-mechanical skills – recognize this theorem as true in its system. In consequence, a human mind outperforms a machine.

Penrose in (1989 [26], 1994 [27]) extended Lucas' reasoning of a positive claim regarding the extra-formal resources available to humans that enable

[4]As reported by Krajewski, Turing believed that even if a machine cannot prove as much as humans can, it is still worth constructing robots.

[5]As reported by Krajewski, Post believed that man cannot construct a machine which can do all the things he can.

them to construct reasoning unavailable to machines. Penrose suggested that in the brain there exists physical basis of non-computable behavior, and he indicated quantum mechanics as a credible candidate. According to him quantum processes might explain not only reasoning of human mathematicians, but also consciousness.

A constructive criticism of Lucas-Penrose style argument got formulated by Putnam (1960 [32], Benacerraf (1967 [1]), Wang (1974 [53]), then later also by Boolos (1995 [2]) or Shapiro (1998 [43]). Penrose's version got criticised in particular by Feferman (1995 [11]), Putnam (1995 [34]) and Shapiro (2003 [44]). Krajewski claims that the ways of criticizing the Lucas-Penrose argument follow one of the two main lines [17, pages 5–6]:

- The mind is a machine and it is consistent, but it cannot prove the Gödel's sentence for itself.[6]

- The mind is a machine, but it is inconsistent, and Gödelian limitations do not apply to that.

Krajewski [17] refutes the Lucas-Putnam argument in yet another way: he observes that iterations of increasingly strong theories proving the corresponding Gödel's sentences can be processed in a purely mechanical or computational manner available to both, humans and machines. In consequence, Krajewski claims that anti-mechanism is not implied by Gödel's incompleteness theorems stand alone. In addition, claims Krajewski, one needs to assume that humans have a privileged access to assessing consistency of the human mind. Krajewski claims that the argument fails because of the necessity of making this extra-formal assumption. This is so, because there is no formal way to account for the formal counterpart of assumptions.

Before I come back, in the last section, to Krajewski's rejection of the anti-mechanism argument, and my proposal how to shift the way of thinking about the reasons for this rejection, I will now focus on the part which is particularly interesting for me, that is the *meta-theoretical* corollary to the anti-mechanism argument stating that humans cannot fully describe out notion of natural number.

[6]This line of argument comes already from Gödel, who distinguished *subjective arithmetic* that humans can do, and who believed that in *objective mathematics* full arithmetic is a consistent theory. He also believed that the concept of computation can be defined without referring to any domain of computation; these claims amount to Gödelian platonism. (Gödel *1951 [13])

2. The concept of natural number and the concept of computation

I initiate my investigation into the nature of the extra-formal elements of the reasoning that enable the conclusion that the human mind is not computable, by discussing the corollary relating human inability to define the concept of natural number. Additionally, I extend the corollary of the claim that humans – for the similar reasons – cannot define the concept of computation. Finally, I present the view according to which the concept of natural number and the concept of computation are closely related.

The fact that every formal definition of the concept of natural number leads to a necessary assumption from the outside of the formal system has been studied in the context of the view in philosophy of mathematics, called *structuralism*. According to structuralism, mathematics is "science of structures", and while defining mathematical objects, one should first target their structural properties. For instance, while defining natural numbers, one should define the structure of natural numbers through relations they hold to each other, and not focus on individual properties of those elements.

Traditionally, structuralism defined natural numbers using the second-order Peano Arithmetic (PA2). PA2 is categorical and the class of (isomorphic) models in which it is satisfied is identified with natural numbers. The usual way of criticising the use of second-order Peano Arithmetic to define natural numbers consists in saying that the underlying logic is "set theory in sheep's clothing" (Quine 1970 [35, page 66]). Second-order logic has the ability, for instance, to express the information that two sets have the same cardinality. The concept of set is itself most frequently (implicitly) defined with a first-order axiomatic theory, such as ZF, that in its turn, is a subject of non-standard interpretations, Löwenheim-Skolem theorem, *etc.*, which makes its intended model "hidden" within a continuum of other non-intended models. Therefore, in order to define the concept of natural number with PA2, humans have two choices. They can get involved in a vicious circle of definitions, or an infinite regression of theorems, or they can use extra-formal resources and admit in an arbitrary manner that there is such a thing as an intended, (or a standard) model of set theory where exists the intended model of arithmetic.

Another, less known, version of structuralism, so called *computational structuralism*, proposes to distinguish the *standard* model of arithmetic from the continuum of non-standard models with the resources of PA1 only (Halbach & Horsten 2005 [14], Quinon & Zdanowski 2007 [36]). In order to

do that, defenders of computational structuralism suggest adding a meta-mathematical constraint regarding the computability of interpretation of functional symbols in the language, and then use Tennenbaum's theorem in order to single out the standard model of arithmetic.

Theorem 2.1 (Tennenbaum 1959). *Let* $\mathcal{M} = \langle \mathbb{M}, +, \times, 0, 1, < \rangle$ *be a enumerable model of* PA1*, and not isomorphic with the standard model* $\mathcal{N} = \langle \mathbb{N}, +, \times, 0, 1, < \rangle$*. Then* $\mathcal{M}$ *is not recursive.*

More explicitly why Tennenbaum's theorem is relevant for the structuralist way of thinking is visible in the transposition of the theorem:

Theorem 2.2 (Tennenbaum transposition). *Let* M *be an enumerable model of first-order Peano arithmetic. If the interpretation of addition and multiplication within* M *are computable then* M *is a standard model for arithmetic (a model with* ω*–type ordering).*

One of the philosophically interesting consequences of the application of Tennenbaum's theorem is that the set of models singled out with its help consists of those ω models, where ω is computable (Quinon & Zdanowski 2007 [36]). Those models are called "intended" and form a proper subset of standard models.

The *intended* model of arithmetic[7], is such a model where functions of addition and multiplication are interpreted as computable functions[8]. Tennenbaum's theorem establishes a connection between a meta-mathematical property of being computable by arithmetical functions, and the order of the elements of the set of natural numbers. Thus, in the most general lines, computational structuralism is a position, according to which the concept of natural number and the concept of computation are closely related.

The usual way of criticising computational structuralism is, again, by pointing out at the vicious circle or infinite regression of definitions that threats the proposed account of natural numbers. The criticism goes as follows: in order to define the concept of natural number, one needs to use the concept of computation, whereas every concept of computation is defined on the domain of (some representation of) natural numbers. Thus, the vicious

[7]Intended models of arithmetic are identified up to a *computable* isomorphism.

[8]The model of arithmetic is intended for both theories PA1 and PA2

circle or the necessity to assume that there is an intended interpretation of what to compute means, or that the intended model of arithmetic is distinguished from within other models.

Analogously, it is pretty straightforward that the concept of computation falls itself into a vicious circle, as in order to account for what "to compute" means, referring, for instance, "to be computed on a Turing Machine", necessitates to account which entities are suitable for computing with (in the case of TM-computations, what can be the input for a Turing Machine). Since the question asked about the input precedes the definition of computing, which is just being given, one cannot use the concept of computing to define which sequences can be used for the input.

More precisely,

> [...] the Church-Turing Thesis states that Turing Machines formally explicate the intuitive concept of computability. The description of Turing Machines requires description of the notation used for the `INPUT` and for the `OUTPUT`. The notation used by Turing in the original account and also notations used in contemporary handbooks of computability all belong to the most known, common, widespread notations, such as standard Arabic notation for natural numbers, binary encoding of natural numbers or stroke notation. The choice is arbitrary and left unjustified. In fact, providing such a justification and providing a general definition of notations, which are acceptable for the process of computations, causes problems. This is so, because the comprehensive definition states that such a notation or encoding has to be computable. Yet, using the concept of computability in a definition of a notation, which will be further used in a definition of the concept of computability yields an obvious vicious circle. (Quinon 2018 [page 338][38]).

In this section, I explained similarities between the process of defining the concept of natural number, the process of accounting for the concept of computation, and the formulation of an anti-mechanism argument based on Gödel's incompleteness theorems. All those contexts are related because the way out of the definitional vicious circles proper to the definitional processes within formal theories, through the necessity of assuming an additional non-formal, meta-theoretical knowledge. In the next section, I will develop on the phenomena of vicious circles and regression ad infinitum.

3. Nested vicious circles

Quinon (2018 [38]) proposes a taxonomy of what can be called "deviant encodings", that is those encodings – or in different words, sequences of symbolic representations of natural numbers – which are non-computable, but which are formally indistinguishable from computable encodings. For instance, in its simplest form the problem presents itself as follows:

> The problem in its purely syntactical version can be formulated as follows. In a definition of Turing computability, one of the aspects that needs to be clarified is the characterization of notation that can be used as an input for a machine to process. If a Turing Machine is supposed to explicate the intuitive concept of computability it is necessary to explain, which sequence of numerals can be used as an input without the use of the concept of computability. That means, we cannot simply say: "sequences that can be used as input are the computable ones" as we have not yet defined what means"to be computable". (Quinon 2018 [38, page 340]).

Deviations refer to non-computable sequences that cannot be distinguished within the general formal context from sequences that are computable and can be used in computations. In this paper, I use the expression "deviant encoding" independently of the ontological framework within which natural numbers are understood. Quinon (2018 [38]) claims that the phenomenon of deviant encodings persist independently of which ontological status we assign to objects of computations (*e.g.*, natural numbers, sequences of symbols, *etc.*).Quinon (2018 [38]) hypothesizes that the phenomenon of deviant encodings persists independently of the philosophical standpoint and provides an analysis of following simplified standpoints: (i) purely mechanical/syntactical approach (nominalism, entwined mathematical concepts); (ii) notations have meanings (mild realism); (iii) semantics comes first (radical realism, platonic insight).

The study of conceptual "deviations" is conducted for a simplified framework where:

- on the syntaxic level there are uninterpreted inscriptions, and where functions are string-theoretical generating strings values from string arguments;

- on the semantic level there are interpretations that can range from the conceptual content ascribed to initially uninterpreted symbols, to Platonic abstract objects, and where functions are number-theoretical sending numbers to numbers;

- between the two levels there is defined a function of denotation.

Deviations occur on each level. Thus, there exist "deviant encodings" deviations that happens on the syntactic level; "deviant semantics" deviations that happens on the semantic level; "unacceptable denotation function" deviations of the denotation function.

The simplified framework is inspired by Shapiro (1982 [42]), who distinguishes string-theoretic functions from number-theoretic functions and searches for "acceptable", that is "non-deviant", ways of associating their domains. The framework is further used by other researchers. Rescorla (2007 [41]) uses it to study behaviour of denotation functions which associate numerals (symbolic representations of natural numbers) to natural numbers (abstract entities) in a non-computable manner. There is a continuum of such mappings.

Differently use the expression "deviant encodings" Copeland and Proudfoot (2010 [7]) for whom the deviations relate encodings, or enumerations, of Turing Machines. The authors claim that a deviant encoding happens when the omniscient programmer "winks at us" to let us know when the number of a Turing Machine (from some standard encoding of Turing Machines), which is being currently processed by some sort of the Halting Machine (a machine computing which Turing Machines stop on an input 0), refers to a machine that stops. In this way, the Halting Machine computes the halting function, which is an uncomputable function. The "wink" of the omniscient programmer gets encoded in the syntactic structure of the numerals: the numerals representing machines that stop, have a special form, – for instance – are even (their general syntactical form can be reduced to "2n" where "n" is any numeral). Copeland and Proudfoot mean by a deviant encoding such a standard enumeration of Turing Machines where the encoding is enriched by an extra-formal feature impersonated by the omniscient programmer (a Turing oracle). This is a specific case of a more general problem where deviant encodings refer to encodings representing natural numbers.

An occurrence of the phenomenon of deviant encodings involving all the levels, is the case of the Semantical Halting Problem (van Heuveln 2000 [52]). Imagine, you have encoded Turing machines with some standard –

computable, thus non-deviant – encoding, and that you believe that symbols have meanings or interpretations. It can happen that even if your syntax is generated in a recursive manner, your semantics is not following any recursive rules. The Halting Machine that processes encodings of Turing Machines is designed to process information on syntax in an algorithmic manner. If inputed with a given non-computable enumeration of Turing machines, the machine will process those non-computable encodings as if it were a standard notation. Again, there is no effective way of defining which semantics are acceptable and which are deviant.

I call "nested vicious circles" the hierarchies of vicious circles that keep reappearing at every stage of syntaxical and semantic complexity of the presented picture.

To give an example of a philosophical position outside the strict theoretical context discussed in this paper, the phenomenon of deviant encodings appears as well in the case of concrete computations.

> In our ordinary discourse, we distinguish between physical systems that perform computations, such as computers and calculators, and physical systems that don't, such as rocks. Among computing devices, we distinguish between more and less powerful ones. These distinctions affect our behaviour: if a device is computationally more powerful than another, we pay more money for it. What grounds these distinctions? What is the principled difference, if there is one, between a rock and a calculator, or between a calculator and a computer? Answering these questions is more difficult that it may seem. (Piccinini 2010 [28])[9].

In (2020 [40]), Quinon notes that the phenomenon of nested vicious circles, relating the concept of computability, do not disappear in the case of explicit inter-definiability between the concept of natural number and the concept of computation, as established by computational structuralism. As I already described it above, the criticism of computational structuralism consists in pointing at the choice between the definitional vicious circles or the necessity of making extra-formal arbitrary assumptions.

The way of extra-formal assumptions is investigated by Button and Smith (2012 [3]) who observed that when the concept "natural number" is expli-

[9]See also Piccinini (2015 [29]).

cated for, the concepts used in this explication, such as "to compute" or "finite" need to be accounted for on their turn, *etc.* In consequence, claim the authors, this problem cannot be tackled by offering more mathematics. An arbitrary decision regarding the meaning of some concept is necessary for the argument from the Tennenbaum's theorem to work. However, as they claim in a slightly undermining way, this is a philosophical problem:

> Suffice it to note that our discussion of Tennenbaum's Theorem illustrates a familiar moral: philosophical problems which are supposedly generated by mathematical results can rarely be tackled by offering more mathematics. (Button & Smith 2012 [3, page 120]).

Dean (2014 [9]) is similarly sceptic when it comes to the purposefulness of using Tennenbaum's theorem to formally single out the standard model of arithmetic. However, differently to Button and Smith, Dean develops a full fledged philosophical position. It is a Putnam-style model-theoretic realism for the concept of computation (see Putnam 1980 [33]). Dean claims that there is no point in trying to find external arguments to distinguish between various standard and non-standard models neither of arithmetic, nor of recursive theory. We should rather use the richness of the model-theoretic universe for studying structural properties of the concept of computation. Dean claims that Tennenbaum's phenomenon shows that there exists continuum of pairs: model of arithmetic and computation in this model of arithmetic. In consequence, the Tennenbaum's result instead of contributing to singling out the standard model of arithmetic, it indicates that there exist non-computable *omega*-models of arithmetic (the so called deviant or weird permutations) with a corresponding concept of computation defined within the model.

The vicious circle faced by computational structuralism, differs from the vicious circles that are the focus of Quinon (2018 [38]). There, I was only concerned by the concept of natural number being indirectly involved in the definition of what "to compute" means. Conceptual structuralism needs to handle a slightly more elaborate idea. Its objective is to explicate the concept of natural number, identified with the standard model of arithmetic. Its solution consists in using the idea that natural numbers, and in particular those which are defined by Peano's axioms, are the entities used for counting and computing. In consequence, natural numbers are defined in terms of computations. However, and this is where the vicious circle arises: one of the characteristic features of the concept of computation is that computation

is *always* defined on some given domain.[10] This domain is always identifiable with the structure of natural numbers. I discuss the nested vicious circles in this context in Quinon (2020 [40]).

4. Conceptual engineering and conceptual fixed points

One of the promising ways out of the impasse consists in embracing that the circularity in the account of what "to be computable" and what "natural number" mean is due to limitations of conceptual analysis. Similarly to other scientific concepts, when analysis is conducted within the strict scope of a given formal theory, one often ends up with a necessity to use the concept which is being defined in the account of some concept used for its definition. Philosophers and logicians see in this feature of conceptual analysis both, an advantage that enables us to understand more about the conceptual structure of the world (see Dean (2014 [9]), and a problem that blocks science from progress (see Maddy (2007 [22]). Rescorla (2007 [41]) identifies problems with conceptual analysis related to the concept of computation. In their paper [3] Button and Smith claim that Tennenbaum's theorem is of no used for a philosopher who wants to distinguishing the standard model from other possible models of arithmetic.

Quinon (2018 [38]) suggests that there is no fully satisfactory way out from vicious circles in definitions, resulting from conceptual analysis. Approaching the concept of computation and the concept of natural number from another methodological perspective, seems to be more fruitful. For instance, in the recent years a particular type of conceptual work gained quite a bit of popularity, it is called *conceptual engineering*. What I try to convey in this section is that the new research on conceptual engineering actually provide additional insight into the possible ways of thinking about the non-mathematical or non-formal knowledge.

According Cappelen (2018 [4]), conceptual engineering is concerned with the assessment and improvement of concepts. As highlight Cappelen and Plunkett (2020 [30, page 3]):

> "since it's unclear and controversial what concepts are (and whether there are any), it's better to broaden the scope along the following

[10] A non-realised Gödel's objective consisted in finding an" absolute" concept of computation, *i.e.*, such a concept of computation that does not depend on any domain.

lines:

> **Conceptual Engineering** = (i) The assessment of representational devices, (ii) reflections on and proposal for how to improve representational devices, and (iii) efforts to implement the proposed improvements."

Researchers involved in developing the methodology of conceptual engineering realised that the method reaches its limits when concepts which are fundamental to the given theory are being scrutinised. They call it "conceptual fixed points". The most extensive reflection has been done in the area of ethics (Cappelen et al. 2020 [5]), but Eklund (2015 [10]) extends it to formal contexts and concepts such as "truth", "belief", or "existence". In addition to traditional arguments used in the ethical contexts, such as "Kantian philosophy [with its regulative ideas], or from a naturalistic philosophy according to which what is innate severely constrains which concepts we can use", Eklund considers basic formal concepts in the spirit of rigid designators.

In moral philosophy, "the moral fixed points" are those moral propositions that are moral truths that always need to be incorporated in a moral system. A normative system which fails to incorporate such propositions is not a moral system, but a normative system of some other kind. The flag example of such a moral fixed point is a proposition "It is wrong to engage in the recreational slaughter of a fellow person" (Cueno & Shafer-Landau 2014 [8]).

Eklund (*e.g.*, 2015 [10, chapter 5]) extends this phenomenon to frameworks outside moral philosophy and, as he calls it, the "thinnest" normative words like "good", "right", "ought". Eklund observes that in each conceptual framework, there exist concepts that are difficult, if not impossible to engineer. "Truth" is one of those concepts. People care about truth, writes Eklund, and they do not care about some conceptually engineered concept "truth*". In consequence, truth is a concept that should keep a fixed position in a conceptual framework, and refer to the natural kin of assertions and beliefs. Similarly, "existence" is a conceptual fixed point. Eklund oposes the claim from in the contemporary meta-ontological debate, where it is assumed "that there are alternative notions of existence that can be employed". He claims that, similarly as in the case of "truth", a conceptual framework that would result from adapting a conceptually engineered concept of "existence" would need to adjust its other key concepts in such a way that the resulting

framework would be isomorphic to the initial one. Thus, "One cannot, so to speak, *selectively* engineer the quantifier".

> Suppose we set out to conceptually engineer truth. Insofar as the job description of truth is that of being the property our beliefs and assertions aim at, the engineering project would be that of finding a property more adequate to that job description. But by what has been noted about Stich's argument, it is hard even properly to conceive of a practice of belief or assertion that is guided by a different property. (Eklund 2015 [10, page 378])

There is one last thing that I consider worth mentioning while talking about conceptual fixed points and mathematical concepts, in particular the concept of computation, that is a possible proximity between conceptual fixed points and fixed points that are traditionally analysed in mathematics in the context of diagonalisation. At the first sight, they do not have much in common[11] as conceptual fixed points relate mostly to the cross-model intended interpretation of a concept, whereas diagonalisation is about self-reference and vicious circles. Conceptual fixed points are concepts interpreted in, what we call in philosophy of mathematics, their intended models. In different words, a fixed point consists of the pair *the engineered concept* corresponding to the intended meaning of the concept, or – to borrow Eklund's expression – the interpretation that "people care about", and *a possible world of interpretation*, which actually corresponds to the intended model of this concept. Both, the concept of natural number and the concept of computation are in this sense conceptual fixed points. A more careful look should be applied to those two phenomena, but in this paper I will just leave it without further comment[12].

5. The Church-Turing Thesis as a Carnapian explication

Another methodological framework that offers a solution for conceptual structure escaping conceptual analysis is the method of Carnapian explication. Quinon (2019 [39]) explores the idea that the structure of the concept

[11] I might be wrong, but I will not try to sort it out in this paper.

[12] If you want to get a more formal description of this phenomenon, you can think of hybrid modal logics which provide a framework for thinking of epistemic access to other possible worlds from the perspective of the selected distinguished world.

of computation, accounted for with the Church-Turing thesis, is the best understood through the method of explication. This section is devoted to the presentation of the method of explication for the concept of computation, and also for the concept of natural number.

Treating the concept of computation, as accounted for in the Church-Turing thesis, as a Carnapian explication has multiple advantages, namely, it overcomes problems of conceptual analysis; it explains how one intuitive concept of what "to be computable" means can be translated into a multitude of extensionally equivalent formal concepts of "to be computable" in a specific formal concept means; it finally provides a ground for thinking of mathematical or formal concepts as "open-textures" evolving through the time (Makovec & Shapiro 2019 [23]); it also relates the initial intuitive pre-scientific concept with the formal concept, because an explication relies on an existing meaning, and offers a specification which suites best possible in a given context.

An explication in the Carnapian sense consists in introducing new formal concepts to the scientific language coined on the basis of everyday concepts. In different words, it is a procedure of transformation of an inexact pre-scientific concept into a scientific one. Moreover, an explication consists in providing a scientific concept within a given context, within an existing theory. It is done in two steps:

- The clarification of the explicatum
- The specification of the explicatum

The rationale for clarification is that a given term may have many different meanings in ordinary language. Unless one of these meanings is clearly picked out from the start and the context of its use is clearly indicated, it is unlikely that the method of explication will yield a useful result. Clarification serves this purpose. As Carnap explains,

> [a]lthough the explicandum cannot be given in exact terms, it should be made as clear as possible by informal explanations and examples. (Carnap 1950 [6, page 3]).

Quinon (2019 [39]) highlights the importance of the clarification stage, the stage which has traditionally been underestimated.

A clarification of the explicandum enables the next step of the explication process, a specification of the explicatum and formulation of the exact concept in the targeted context.

Since most often several clarifications can be foreseen, and several scientific contexts are available, one pre-scientific concept can be explicated in various manners. In order to decide which explication is the most successful, Carnap proposes four criteria that can be applied for assessing the value of an explication, and also for comparison between available options.

- *Similarity to the explicandum* most of the cases in which the explicandum has so far been used, the explicatum can be used; however, close similarity is not required, and considerable differences are permitted.
- *Exactness* the rules of use of explicatum have to be given explicitly and precisely, for example, by providing a concept with the formal definition.
- *Fruitfulness* shall be "useful for the formulation of many universal statements".
- *Simplicity* an explication should be as simple as the previous three allow it.

I think that it is worth investigating whether abandoning the path of analysis and taking the path of explications could offer an additional insight into the conceptual structure of formal concepts, and also informal concepts lying in the foundations of their formalization. The idea that every formal concept is, – at least in subjective arithmetic (to borrow Gödelian terminology) – grounded upon, or issued from, everyday intuitive, pre-scientific concept. The next section, is devoted to a preliminary investigation into a possibility of extending the idea that the method of explication, consisting in building up the formal concept out of the intuitive concept, is anyhow relevant to the anti-mechanism argument against the computability of mind using Gödel's incompleteness theorems.

Both intended interpretations determined in consequences of accepting conceptual fixed points solution and the choice of the formal aspect and the formal context at the stage of the concept clarification in the process of Carnapian explication, share a similar threat. In the case of fixed point solution and in the case of clarification an agent needs to take an arbitrary decision regarding the intended interpretation.

6. Theory of mind and computations

In this section, I propose an additional complication to the method of Carnapian explication, which is a temporary, or a phylogenic, aspect of conceptual development.

The method of Carnapian explication enables introducing new formal concepts to the language by transforming an intuitive pre-scientific concept into a new scientific concept within some formal concept. Usually, at the stage of clarification one chooses the formal meaning that the formalisation of the intuitive pre-scientific concept and also the targeted formal concept. What I propose in this section is an additional dimension to the clarification stage: a relativisation to the phylogeny of the targeted concept. A the stage of clarification, in addition to deciding which aspect of the intuitive concept one wants to formalise one needs to realise that each concept develops. The phylogenic development of the concept of natural number and the concept of computation is studied in Shapiro on open texture (2013 [45]).

The relation between the concept of computation and the concept of natural number underwent a very dynamic development. In consequence, grew the set potential clarifications of intuitive concepts of computation and of natural numbers. What is interesting from my perspective is that computability is today an expected feature of natural numbers. Natural numbers are those mathematical entities that are all days long used for enumerating and computing, for programming, and in various sorts of logistic projects as an underlying discrete structure. Both concepts get increasingly important in the everyday life of our society. This is called digitalisation.

Various areas of digitalisation are additionally reinforced by the fact that computationalism – even if its formal details are still discussed by philosophers, mathematicians and logicians – is today the mainstream theory of mind. This process is described by Turkle (1984/2004, 2011, 2015) who studies how concepts from computer sciences and robotics get into common language and how they change ordinary people's approach to inter-personal relations or ethical questions.

According to Turkle the intensity in which digitalisation of the everyday life develops is strongly connected to the fact that computational language was first used to reformulate our perception of our own mind and our consciousness.[13]

[13]Turkle's earlier work related similar development of conceptual trends in explanation

When Turkle speaks about her experience with the digitalised society, she compares the two experiences:

> My experience at MIT impressed me with the fact that something analogous to the development of a psychoanalytic culture was going on in the worlds around computation. At MIT I heard computational metaphors used to think about politics, education, social process, and, most central to the analogy with psychoanalysis, about the self. (Turkle 1984 [49, page 305])

She sees in it a first step in the cultural assimilation of a new way of thinking:

> The essential question in such work is how ideas developed in the world of high science are appropriated by the culture at large. In the case of psychoanalysis, how do Freudian ideas move out to touch the lives of people who have never visited a psychoanalyst, people who are not even particularly interested in psychoanalysis as a theory? In the study of the nascent computer culture, the essential question was the same: how were computational ideas moving out into everyday life? (Turkle 1984 [49, page 305]).

She searches how "the idea of mind as a program enters into peoples sense of who is the actor when they act". A model of mind that is adapted by the society influence how people think about their frustrations and disappointments, their relationships with their families and with their work [23, page

of phenomena of everyday life that had place in France in the 1960s and 1970s in consequence of the spread of psychoanalytical ideas, see her book "Psychoanalytic Politics: Jacques Lacan and Freud's French Revolution" from 1978 [48]). In "The Second Self: Computers and the Human Spirit" (1984/2005 [49]) Turkle describes these changes that gets into general culture from the digitalisation and robotics in the same way as "psycholanalytic culture" penetrated structures of the general social and political life in France: "Psychoanalytic language spread into the rhetoric of political parties, into training programs for schoolteachers, into advice-to-the-lovelorn columns. I became fascinated with how people were picking up and trying on this new language for thinking about the self. I had gone to France to study the psychoanalytic community and how it had rein- vented Freud for the French taste, but I was there at a time when it was possible to watch a small psychoanalytic community grow into a larger psychoanalytic culture." (Turkle 1984 [49, pages 304–305]).

305]. On the other hand, says Turkle, computer became a new constructed object - "a cultural object that different people and groups of people can apprehend with very different descriptions and invest with very different attributes. Ideas about computers become easily charged with personal and cultural meanings" (Turkle 1984 [49, page 308]).

In her other books, Turkle studies human attachment to objects. In the volume of essays "Evocative Objects: Things We Think With" she speaks about attachment that people, many of her friends, developed with physical objects. In her book, "Alone Together" Turkle (2011 [50]) extends her observations to different types of automated artificial agents, such as virtual agents mediated by electronic support, or robots. In the series of social experiments, where she asks her subjects to interact with an automated artificial agent, she observes that the stronger attachment develops in the most vulnerable members of our society, such as neglected children with unfulfilled emotional needs, or as old people suffering from the lack of human interactions. Our natural inclination to form emotional attachment with humans, and with objects in the absence of humans, might soon lead to even more human-AI interactions. Those interactions are obviously structured in a very particular, very automated, way, which even more strongly influence digitalisation of the language we use.

Krajewski makes a similar observation in the last section of the paper.

> Our attitude toward the arguments of Lucas, Penrose, and others is shaped mostly by our general vision of machines and minds. And this vision adjusts with changes of civilization. For the youth of today, if I may judge from listening to my students, our computerized world makes it easier to accept the idea that anything is mechanizable – including the mind. [17, page 32]

I propose a hypothesis that at least a part of the confusion regarding the specificity of the conceptual structure of the concept of computation contributes to the confusion regarding nature of human reasoning and the human mind. In consequence, I claim that at least partially the "feeling" that there exist non-computational processes come from the complexity of the conceptual structure of the concept of computation.

7. The Lucas-Penrose argument and extra-formal concepts

Let me now come back to the anti-mechanism argument against computability of mind based on Gödel's incompleteness theorems. In this section, I reconstruct Krajewski's argument that in order to render the anti-mechanism argument work, one needs to add an extra-formal assumption regarding the consistency of the underlying theory, that is, the theory corresponding to the human mind. I present the core of Krajewski's criticism: it is not possible to formalize the extra-formal assumption and hence, the whole Lucas' argument is fallacious. I disagree with Krajewski's claim that a formalization of the extra formal assumptions is not possible. Thre are contemporary philosophical methods that might enable formulation of such a formalization. In previous sections, I shown how the methodological and conceptual framework issued from Carnapian explications. Instead, I focus on another problem which issues form an internal characteristic of formal contexts, namely on this aspect of the argument which leads to a circular reasoning in the reasoning, *i.e.*, in order to show that the human mind (T_{HM}) outperforms a machine (T_M) one needs to assume that the human mind is consistent and knows it (and in this way outperforms a machine that can never "know" if it is consistent or not). Again, I already discussed how the method of conceptual engineering enable structuring thinking of extra-formal assumption and the resulting circular reasoning.

In the second part of this section, I continue my investigation of possible extra-formal assumptions relative to the anti-mechanism argument based on Godel's incompleteness theorems.

The Lucas' anti-mechanism argument based on Gödel's incompleteness theorems consists of two parts. Firstly, Gödel's results establish that each sufficiently rich consistent theory admits a Gödel's sentence and also that none such theory can prove its own consistency.

Let T be a consistent theory containing arithmetic, let φ_T be the Gödel's sentence for the theory T.

$$Con(T) \Rightarrow T \nvdash \varphi_T$$

$$Con(T) \Rightarrow T \nvdash Con(T)$$

Moreover, it is broadly known that an inconsistent theory proves any sentence, but Gödel's incompleteness theorems do not apply to an inconsistent theory.

Secondly, a human mathematicians can work with subsequent increasingly stronger theories,

$$T_1 = T \cup Con(T)$$
$$T_2 = T_1 \cup Con(T)$$
$$\vdots$$
$$T_{n+1} = T \cup Con(T)$$

which – for some defenders of an anti-mechanism argument – signifies that human mathematicians outperform machines. Krajewski objects this view claiming that the construction of the hierarchy can be fully mechanised. In consequence, he claims that the ability to construct and work with the hierarchy of increasingly stronger theories alone is not sufficient for formulating the anti-mechanism argument. As stated by Krajewski [17, page 30–31], additional assumptions are missing.

> In addition to Gödel's results, at least two assumptions that are not self-evident are used in the above reasoning. First, every exact proof of our consistency can be formalized, second, it is possible to express "our consistency." [...] If this is accepted, one could question the second point. It is not clear at all how one can express "our consistency." Basically there are two options to express this: either (i) by the common sense statement "I am consistent" or (ii) by a formal counterpart to this statement. Let us consider them in turn.
>
> In case (i) we refer to a common sense statement, which have no connection to formal considerations. Hao Wang [in 1974, pages 317-320] reflected on just this statement and believed that it is not provable. [...] If that were possible, it would mean that we are not machines, or that we are not even equivalent to machines in the realm of proof-producing reasoning. We certainly may believe that, but it is no more than a general feeling.
>
> In case (ii) we consider the formal counterpart to a loose statement expressing consistency; [...] The usual meaning of the statement refers to the will to avoid contradictions, to the reliability of our vision of the world, to the claim that the methods used by mathematicians are unfailing. The sentence $Cons$ or any

> other similar arithmetical formula is rather far from those ideas. Thus, while something is strictly proved, it is unclear to what extent the conclusion conveys our consistency.

Krajewski's reasoning can be reconstructed as follows. Applying the formal predicate "being consistent" can only apply to a formal theory. Applying the formal predicate "being consistent" to anything else than a formal theory is a categorical mistake. In consequence, if consistency is to be predicated of the human mind, the mind must have certain formal properties and needs to be identified with a theory. There exist the following options:

- If human mind is a theory and it is consistent, then as to all other theories, a Gödel's sentence applies to it and the human mind encounters the same constraints as any theory (a machine).

- If the human mind is a theory and it is inconsistent, then Gödelian argument limitations do not apply at all.

If the human mind is a theory, a human disposing of a mind cannot know – from the formal point of view – if it is consistent or not. In consequence, in order to prove that the human mind outperforms a machine, a second extra-formal additional assumption needs to be made. It has to be assumed that the human mind is indeed consistent. This assumption can be done in one of the two ways. "Case (i)", "I am consistent" cannot be formalised. "Case (ii)", there exists a formal counterpart of "I am consistent".

My analysis of "case (i)" is in line with the analysis of Krajewski. If "I am consistent" is an informal statement, it is useless for any formal proof. And here we speak of being able to *prove* more than a machine. Whereas Lucas' argument is supposed to be a formal proof of the superiority of the human mind over a machine.

My analysis of "case (ii)" differs from Krajewski's analysis. His argument resumes to the idea that each formalisation of the informal "I am consistent" remains – maybe more informed or more precise – but still an informal account. As such it is useless for any formal proof. I think that conclusion from (ii) is different. An agent *can* find a formal counterpart of the statement "I am consistent", or rather "the theory constituting my mind is consistent". The framework of Carnapian explications enables us to understand how it can be done.

I also assume that an agent *can* recognise her own consistency. This insight is available to a human being, while it is – on the ground of the second Gödel's incompleteness theorem – unavailable to a machine. This extra-formal assumption is necessary for formulating an anti-mechanism argument against computability of mind. This is also exactly at this point where a vicious circle occurs. We are just being in the act of proving that the human mind outperforms a machine, and here we are assuming exactly this.

Another possible extra-formal assumption that can be made to enable the anti-mechanism argument based on Gödel's incompleteness theorem, is the ability to refer to the intended model of arithmetic[14]. Instead of assuming that the human mind is consistent, that means, assuming that the theory underlying all human reasoning is a consistent theory which does not prove both a φ and a $\neg\varphi$, for every φ, in order to use Gödel's incompleteness theorems to support anti-mechanism argument, one can assume that the human mind is able to refer to the intended model of arithmetic. The assumption that the human mind can refer to the intended model of arithmetic disables the possibility that the Gödel sentences get to have non-standard Gödel's numerals.

In the way it is usually interpreted – in particular in the context of philosophical argumentation supporting anti-mechanism argument that the human mind is non-computable – Gödel's incompleteness theorems provide us with the information from the perspective of a formal system. The semantical aspect is taken for granted. When the model theoretical reasoning is applied, Gödel's incompleteness theorems indicates that there exist non-standard models where the (non-standard) Gödel number of the proof of Gödel's incompleteness theorems have its (semantical) reference. It also means that there exist models where the Gödel (non-standard) number of the proof of the negation of Gödel's first theorem, has an interpretation as a (non-standard) natural number.

What is famously referred to by Gödel's platonism is his belief that there is a model of arithmetic in which all arithmetical truths are satisfied. This is obviously not the intended model of arithmetic that humans have a privileged cognitive access to, but the model of arithmetic in the objective mathematics

[14]The intended model is intended for both $PA1$ and $PA2$ and for this reason I do not make a distinction between the intended model of $PA1$ and the intended model of $PA2$. I can think of a philosophical position that makes such a distinction, but for my purpose that would unnecessarily complicate my presentation.

(Gödel *1951 [13]).

Conclusions

Additional to the critical analysis of Krajewski's rejection of the anti-mechanism based on Godel's incompleteness theorems to which I suggest some possible improvements, my paper is sympathetic to the idea that certain key concepts in formal contexts naturally fall into circular or infinite reasonings. In this way, I try to shift attention from the theory of the human mind and consciousness, to the study of the conceptual structure of the language.

In my paper, I explored similarities between various formal contexts in which key concepts fall into a vicious circle of reasoning. I looked at the formalisation of the concept of natural number, of the concept of computation, and at the concept of consistency in the context of Gödel's incompleteness theorems. I suggested that the way to switch from an informal pre-scientific concept to a full-blooded formal scientific concept formulated in an adequate formal context is best modeled by Carnapian explications. I also suggested that the phenomenon of conceptual fixed points offers a methodological framework to think of intended interpretations necessary to jump out of circularity.

References

[1] Benacerraf, P. (1967). God, the Devil, and Gödel. Monist 51: 9-32.

[2] Boolos, G. (1995). Introductory note to *1951. In: Gödel, K., *Collected Works, Volume III, Unpublished essays and lectures*, Feferman S., et al. (eds.), Oxford University Press (1995): 290—304.

[3] Button, T., Smith, P. (2012): The Philosophical Significance of Tennenbaum's Theorem. Philosophia Mathematica 20(1): 114–121.

[4] Cappelen, H. (2018). Fixing Language: An Essay on Conceptual Engineering. Oxford University Press.

[5] Cappelen H., Plunkett D. & Burgess A. (eds.) (2020), Conceptual Engineering and Conceptual Ethics. Oxford University Press.

[6] Carnap, Rudlof (1950), Logical Foundations of Probability. Routledge and Kegan Paul.

[7] Copeland, J. & Proudfoot, D. (2010). Deviant encodings and Turing's analysis of computability. Studies in History and Philosophy of Science 41: 247–252.

[8] Cuneo T. & Shafer-Landau R. (2014). The moral fixed points: new directions for moral nonnaturalism. Philosophical Studies 171: 399-443.

[9] Dean, W. (2014), Models and Computability. Philosophia Mathematica 22 (2): 143–166.

[10] Eklund, M. (2015). Intuitions, Conceptual Engineering, and Conceptual Fixed Points. In: Daly C. (ed.) The Palgrave Handbook of Philosophical Methods. Palgrave Macmillan: 363-385.

[11] Feferman, S. (1995). Penrose's Gödelian argument. Psyche: An Interdisciplinary Journal of Research on Consciousness.

[12] Gödel, K. (193?), Undecidable Diophantine Propositions. In: Gödel, K., *Collected Works, Volume III, Unpublished essays and lectures*, Feferman S., et al. (eds.), Oxford University Press 1995: 164–175.

[13] Gödel, K. (*1951). *1951 is Gödel's 1951 Gibbs lecture. Some basic theorems on the foundations of mathematics and their implications, lecture manuscript. In: Gödel, K., *Collected Works, Volume III, Unpublished essays and lectures*, Feferman S., et al. (eds.), Oxford University Press 1995: 304—323.

[14] Halbach, V. & Horsten, L. (2005). Computational Structuralism. Philosophia Mathematica 13(2): 174–186.

[15] Hofstadter, D.R. (1979). Gödel, Escher, Bach, and Eternal Golden Braid. Basic Books.

[16] Krajewski, Stanislaw (2007). On Gödel's Theorem and Mechanism: Inconsistency or Unsoundness is Unavoidable in any Attempt to 'Out-Gödel' the Mechanist. Fundamenta Informaticae 81: 173–181. Reprinted in Topics in Logic, Philosophy and Foundations of Mathematics and Computer Science: In Recognition of Professor Andrzej Grzegorczyk (2008).

[17] Krajewski, Stanislaw (unpublished manuscript). On the Anti-Mechnist Arguments Based on Gödel's Theorem.

[18] Lucas, J.R. (1961). Minds, Machines and Gödel. *Philosophy* 36 (137): 112-127.

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

[20] Lucas, J.R. (1990). A paper to read to the Turing Conference at Brighton on April 6th, 1990. Unpublished manuscript http://users.ox.ac.uk/ jrlucas/Godel/brighton.html

[21] Lucas, J.R. (1996). Minds, Machines and Gödel: A Retrospect. In: Millican P. & Clark A. (eds.). Machines and Though. Oxford University Press.

[22] Maddy P. (2007). Second Philosophy. A Naturalistic Method. Oxford University Press.

[23] Makovec, D. & Shapiro S. (eds.) (2019). Friedrich Waismann. The Open Texture of Analytic Philosophy. Springer.

[24] Nagel, E. & Newman J.R. (1958). Gödel's Proof. New York University Press.

[25] Nagel, E. & Newman J.R. (1961). Answer to Putnam. Philosophy of Science. 28: 209–211.

[26] Penrose, R. (1989). The Emperor's New Mind: Concerning Computers, Minds and The Laws of Physics. Oxford University Press.

[27] Penrose, R. (1994). Shadows of the Mind: A Search for the Missing Science of Consciousness. Oxford University Press.

[28] Piccinini, G. (2010/2017). Computation in Physical Systems. In: Zalta, E.N. (ed.) The Stanford Encyclopedia of Philosophy. Metaphysics Research Lab, Stanford University.

[29] Piccinini, G. (2015). Physical Computation: A Mechanistic Account. Oxford University Press.

[30] Plunkett D. Cappelen H. (2020). A Guided Tour Of Conceptual Engineering and Conceptual Ethics. In: Cappelen H., Plunkett D. & Burgess A. (eds.), Conceptual Engineering and Conceptual Ethics. Oxford University Press (2020): 1–26.

[31] Post, E. (1941). Absolutely Unsolvable Problems and Relatively Undecidable Propositions – Account of an Anticipation. In: Davis, M. (ed.). The Undecidable (1965). Raven Press

[32] Putnam, H.(1960). Minds and Machines. In: Hook S. (ed.). Dimensions of Mind. A Symposium. Collier-MacMillan.

[33] Putnam, H. (1980). Models and Reality. Journal of Symbolic Logic 45(3): 464–482.

[34] Putnam, H. (1995). Review of The Shadows of the Mind. Bulletin of the American Mathematical Society. 32(2): 370–373.

[35] Quine, W.V.O. (1970). Philosophy of Logic. Harvard University Press.

[36] Quinon, P. & Zdanowski, K. (2007). Intended Model of Arithmetic. Argument from Tennenbaum's Theorem. In: Cooper, S. Barry; Kent , Thomas F.: Löwe, Benedikt & Sorbi, Andrea (eds.), Computation and Logic in the Real World, CiE: 313–317.

[37] Quinon, P. (2014), From Computability over Strings of Characters to Natural Numbers. In: Olszewski, A.; Brożek, B. & Urbańczyk, P. (eds.), Church's Thesis, Logic, Mind & Nature. Copernicus Center Press: 310–330.

[38] Quinon, P. (2018). Taxonomy of Deviant Encodings. In: Manea F., Miller R. & Nowotka D. (eds) Sailing Routes in the World of Computation. CiE 2018. Lecture Notes in Computer Science 10936 Springer: 338-348.

[39] Quinon, P. (2019). Can Church's Thesis be Viewed as a Carnapian Explication?, Synthese: Online First.

[40] Quinon, P. (2020). Implicit and explicit examples of the phenomenon of deviant encodings. Journal of Studies in Logic, Grammar and Rhetoric. In press.

[41] Rescrola, M. (2007), Church's thesis and the conceptual analysis of computability. Notre Dame Journal of Formal Logic 48 (2): 253–280.

[42] Shapiro, S. (1982). Acceptable Notation. Notre Dame Journal of Formal Logic 23(1): 14–20.

[43] Shapiro, S. (1998). Incompleteness, mechanism, and optimism. Journal of Philosophical Logic 4: 273–302.

[44] Shapiro, S. (2003). Mechanism, truth, and Penrose's new argument. Journal of Philosophical Logic 32: 19–42.

[45] Shapiro, S. (2013). Computability, Proof and Open-texture. In: Olszewski A., Wolenski J., Janusz R., Church's Thesis After 70 Years, Walter de Gruyter: 420-455.

[46] Turing, A. (1936). On Computable Numbers, with an Application to the Entscheidungsproblem. Proceedings of the London Mathematical Society 42: 230–265; correction in (1937) 43: 544–546; reprinted in (Davis 1965: 115–154); page numbers refer to the (1965) edition.

[47] Turing, A. (1950). Computing machinery and intelligence. Mind: 433–460.

[48] Turkle, S. (1978). Psychoanalytic Politics: Jacques Lacan and Freud's French Revolution. Basic Books.

[49] Turkle, S. (1984/2005). The Second Self: The Second Self: Computers and the Human Spirit. MIT Press.

[50] Turkle, S. (2011). Alone Together: Why We Expect More from Technology and Less from Each Other. Basic Books.

[51] Turkle, S. (2015). Reclaiming Conversation: The Power of Talk in a Digital Age. Penguin Press.

[52] van Heuveln, B. (2000), Emergence and consciousness: Explorations into the Philosophy of Mind via the Philosophy of Computation. Ph.D. thesis. State University of New York at Binghampton.

[53] Wang, H. (1974). From Mathematics to Philosophy. Routledge and Kegan Paul.