Borderless Philosophy 2 (2019): 117-172.

Proteons: Towards a Philosophy of Creativity

Andreas Keller

Preface

Vita brevis, ars longa.

—Ascribed to Hippocrates

This essay is a sketch. There are many loose ends left. In part, this sketchiness is a consequence of my limited time, but it is also, as we will see, a consequence of the nature of the subject.

I hope that in the future I will have time to pursue further some of the strands of thought only hinted at here, and some that remained left out. However, although I am not yet old, I am also no longer young. If I won't be able to work out many of the further details myself, I hope that others will take up this task. This work will, however, always remain piecemeal and there will always be several possibilities to develop these thoughts further. I myself will probably not be able to go very far along this road. So I dedicate this essay to the young people, and especially to my daughter Nange and to my nephew Yata.

My special thanks go to my long-standing friend Kurt Ammon, who paved this way and walked it before me, 40 years ago, and upon whose thoughts I am building here. Some of the formulations in the following text originate from him and I didn’t always have the time to mark these exactly. His thought has influenced my own so profoundly, by reading and proof-reading his publications and by personal discussions, that in some cases, I don’t know which ideas are originally his and which are my own; so one big general reference goes to his works.

Thanks also to those who provided helpful comments on blog articles where some of these ideas have been presented before and to the reviewers for their comments and corrections.

Abstract

In this essay, it is argued that physical reality, the human beings facing it, the totality of human knowledge and cognition, human culture, language, and science cannot be described completely. Each formal theory about such entities is incomplete. For such entities, the term “proteon” is introduced. Some properties of proteons are discussed, especially different forms of incompleteness or anomalies. Referring to the work of

Kurt Ammon, cognitive proteons (called "creative systems" by Ammon) are discussed. The notions of creativity and of analytical spaces are introduced. Creativity is defined as the ability of a cognitive proteon to move out of the scope of any formal theory about it. Analytical spaces form the components of cognitive proteons or their knowledge. They can be viewed as pieces of knowledge that can be described as formal theories or algorithms. Since they develop in interaction with the objects they refer to, they cannot be completely separated from these objects. The application of the idea of proteons in different areas (mathematics, physics, biology, theory of science, cognitive science and AI, linguistics, philosophy, pedagogy, practical philosophy, aesthetics, human life in general) is sketched, although in most cases only in a very preliminary manner. In a more technical section, some results relevant for AI and cognitive science are treated in a slightly deeper way. Proofs are sketched that indicate that formal theories of cognition are incomplete. In a Concluding Summary, the main points of the paper are extracted.

1. Knowledge and Reality

1.1 The Development of Knowledge

For we know in part and we prophesy in part…

—1 Corinthians 13:9

Knowledge arises from our interaction with reality, including ourselves and our previous knowledge. When we look at the totality of our knowledge, as it is stored in our books, our book shelves and libraries, in our computers and on the internet, and of course, in our brains, what we see is that the amount of knowledge we have is increasing more and more and that it is branching and developing. It looks like reality is inexhaustible, so we never reach the end. Our knowledge is never complete.

Perhaps this is a transient state, and at some point in history, our scientists and scholars will be able to declare knowledge to be finally complete, quit their jobs, and go on pension. And we will finally have a complete theory of everything that can answer every question. But, well, honestly, this does not sound very plausible. The process of generating knowledge might come to an end one day because of the collapse of our civilization, but not because we will run out of research topics.

Please note that I do not use the term "knowledge" here in the sense of "justified true belief." I consider "justified true belief" to be a theoretical abstraction that is of no practical use whatsoever. We may justify our beliefs, but we will never be able to be sure what part of the totality of what we believe is really true, and every justification might be called into question. Some parts of our beliefs might actually be true, but we will never know."[1]

Instead, when I say "knowledge," I mean real knowledge, in a sense much closer to the everyday meaning of the term. I mean that stuff in our libraries or on the internet. There are gaps in it, there are inconsistencies, there are errors, there is vagueness, and there are different degrees of justification, or lack thereof. But our knowledge can be extended, revised, corrected, and made more exact and more explicit. It is developing. It is dynamic. While we are revising and developing our knowledge, our language develops with it. By "language" I mean both what is—perhaps wrongly—called "natural" language as well as any extension of it, like, for example, diagrams and tables and charts, mathematical and chemical formulas, database systems and graphics file formats, emojis, etc. [2]

Knowledge develops out of our interaction with reality. It looks like reality cannot be described completely, it seems to be inexhaustible. All our theories (i.e., systems of knowledge) that we develop about it seem to be *incomplete*. There are, of course, special domains for which our knowledge may be complete, but these are only subsections of reality, so the knowledge about them is special, i.e., incomplete with respect to the totality of reality. Reality always has more properties than can be derived in any of our theories about it. In other words: reality will always *surprise* us.

---

[1] It will become clear in the following that I do not share a Kantian point of view. While at any given time, one might be able to identify some kind of framework of basic cognitive structures, I believe this framework is developing historically. I side with Dilthey in this respect, who wrote (Dilthey 1983, p. 44):

> Das a priori Kants ist starr und tot; aber die wirklichen Bedingungen des Bewußtseins und seine Voraussetzungen, wie ich sie begreife, sind lebendiger geschichtlicher Prozeß, sind Entwicklung, sie haben ihre Geschichte, und der Verlauf dieser Geschichte ist ihre Anpassung an die immer genauer induktiv erkannte Mannigfaltigkeit der Empfindungsinhalte.

Translation:

> Kant's a priori is rigid and dead; but the real conditions of consciousness and its premises, as I understand them, are living historical process, are evolution, they have their history, and the course of this history is their adaptation to the ever more precisely inductively recognized multiplicity of sensory contents.

[2] The terms "description," "(formal) theory," "algorithm" and "knowledge" do not imply that we are dealing with conceptual descriptions only. Conceptual knowledge is a special case. To use a term from computer programming, our thought processes can employ all kinds of different data types. Moreover, new kinds of representations can be invented. Conceptual representations and conceptual thought processes are just one example of such different "formats of cognition."

One basic thesis of this essay is that reality is indeed inexhaustible in this sense. Any theory about reality is incomplete. It is possible to extend theories and replace each one with a more comprehensive one. But that one will be incomplete in turn.

A similar situation is well known in mathematics. Mathematical objects can be constructed for which the incompleteness of all formal theories about them can be proved. This has been known at least since Kurt Gödel published his 1931 paper "Über formal unentscheidbare Sätze der Principia mathematica und verwandter Systeme." The incompleteness results of mathematics mean that there are entities in mathematics that cannot be described completely in terms of any formal theory about them. As paradoxical as it might sound, mathematics, that most formal of all activities, cannot be formalized completely. Each incomplete theory can be extended, i.e., replaced by a more comprehensive one in which the gap of the previous theory is patched, but a gap will open up in that extended theory as well.

A tacit assumption of scientistic philosophy seems to be that a complete theory of everything is possible. However, if we can conceive of mathematical entities that cannot be described completely in terms of any single formal theory, then there is no justification for such an assumption. It might be true, but it is just a hypothesis. Reality might be non-formalizable and the laws of physics might be non-computable in the sense of computability theory.

Reality might be something that cannot be described completely in terms of any formal theory. Now, "something that cannot be described completely in terms of any formal theory" or "something for which every formal theory about it is incomplete" is a bit long.

1.2 Introducing a New Term

The limits of my language mean the limits of my world.

—L. Wittgenstein, *Tractatus Logico-Philosophicus*, prop. 5.6

There is a gap in our language here and I want to try to close it by suggesting a new philosophical term. So what I am looking for is a term for an entity that cannot be described completely by any single formal theory (or algorithm – the two notions are essentially equivalent, see below), something for which a complete and at the same time exact and explicit description is impossible in principle or at least for all practical purposes, e.g. for reasons of (computational) complexity.

It seems to me that human beings as well as their societies and cultures are such entities (and I am going to provide some arguments for this opinion below), but

I think many physical systems are such entities as well. Actually, the word "system" does not really fit here because it has a connotation of something systematic, i.e., something that can be captured completely and exactly by some theory. Scientists use the term "system" as a very general term for practically everything, as a scholarly synonym for "thing,", and just assume that everything is a "system." There seems to be a tacit (and I suppose: wrong) assumption in science that everything is formalizable, and this assumption seems to be hidden right inside the term "system. Instead of using a contorted circumscription like "something for which every formal theory of it is incomplete" and instead of using a negative term derived from another word by means of "non-," "in-," "un-," or "a-,", such as "non-formalizable,", "informal," etc., we should have a *simple positive* term.

Ancient mythology has often provided the material for new words and terms in science and philosophy, so looking for a fitting new philosophical term, I came upon Proteus. An ancient god of the seas, Proteus has the ability of changing shape. He or she or it does not even have any primary, "real," definite shape. The idea of such a god might have arisen from the experience of sailors with the unpredictability of the weather and the seas. Modern science has made some progress in the area of weather forecasts, but there is reason to believe that an exact long term forecast of the weather will always remain impossible (so here we have a prime example from science for the kind of entity for which I am trying to find a term).

In an article about Proteus, Wikipedia tells us:

> He can foretell the future, but […] will change his shape to avoid having to.[…] From this feature of Proteus comes the adjective protean, with the general meaning of "versatile", "mutable", "capable of assuming many forms". "Protean" has positive connotations of flexibility, versatility and adaptability.[3]

So I want to suggest deriving the term we are looking for from the name "Proteus." Correspondingly, by the word

proteon

I mean something—an object (either mathematical or physical) or a process—that cannot be described completely by any single formal theory or algorithm. This implies that each formal (i.e., exact and explicit) description of a proteon is incomplete, although the possibility to extend or improve any given description of it might exist. A proteon is non-formalizable by definition. There might be different

[3] See Wikipedia, "Proteus," (2018), available online at URL = <https://en.wikipedia.org/w/index.php?title=Proteus&oldid=857609946>.

reasons why something cannot be described completely but I would like to keep the meaning of the term a little bit vague here, in order to keep it flexible and protean. So there might be several different types of proteons that have this property for different reasons. The term may also be used for cases where a complete formal description might be possible theoretically ("in principle") but where it is practically impossible, e.g., for reasons of complexity (e.g. of computational complexity). I suggest using the plural form "proteons."

Moreover, in the context of this terminology, I would use the (already existing) adjective

protean

to refer to something that is a proteon.

A proteon is *not* a system. According to the terminology I am proposing here, the term

system

is the opposite of the term "proteon," or in other words, a system is something for which a complete and exact description is possible. I would then, of course use the word

systematic

as the adjective associated with "system."

I hope that these terms will make it easier to think about the non-formalizable and that they extend what can be thought about easily. The protean aspect of reality seems to be somehow hidden to many people until now, so introducing these simple terms might extend what we can think about.[4]

[4] In sufficiently high magnification and resolution, to use a metaphor from microscopy, everything might be a proteon. Systems are idealizations, "as-if-structures," where certain aspects are left out. For example, a house can be viewed as a system, described by the blueprints of the architect or the knowledge of the inhabitants about it. If you leave the house to itself for long enough, however, it will go its own way, i.e., the "underlying" physical entity that "emulated" the house-system has properties not covered by the systematic description and these will start to show: the house "breaks" or "decays," but the underlying physical entity does not; for this entity, the decay processes are normal processes like anything else.

2. Basic Concepts

2.0 Introduction

The following section contains some of the central ideas about proteons, and also provides some central insights into the theory of proteons. As much as possible, I am attempting here to present these ideas in a form that can be grasped without much of a background in mathematics or computer science, but a certain level of technical difficulty is unavoidable in some places.[5]

2.1 Formal Theories and Algorithms

In this essay, I will often write about "algorithms" and about "formal theories." These two concepts can be viewed as largely equivalent. For any formal theory, an algorithm can be constructed that enumerates all statements derivable in that theory. Conversely, for every algorithm (roughly speaking, a "program" or "app"), a formal theory can be constructed in which one can derive all statements saying that the algorithm is calculating a certain result for a given input or set or sequence of inputs. So in this essay, I am using both terms interchangeably since both notions have the same "expressive power."

2.2 Computability Theory and Complexity Theory as Core Disciplines of Philosophy

The branch of mathematics that deals with the computability of functions, i.e., with the question what can or cannot be done by means of algorithms, is known as computability theory. Its "little sister" is complexity theory, dealing with the question which functions can be effectively calculated with a given amount of resources (time and information storage capacity) and which ones are practically non-computable because they would need resources exceeding whatever can be made available by any technology (i.e., functions that are computable "in principle" in the sense of computability theory, but not computable in practice). It has been noted above that the notions of algorithm and of formal theory are equivalent. The theory of computability, together with the theory of (computational) complexity thus deals with the question which types of entities can be formalized and which cannot. We can therefore also call it the theory of (non-)formalizability. As such, I consider it a centerpiece of epistemology and thus of philosophy as a whole, since it deals with the limits of formalizations, as used in systematic science and analytic philosophy. So the topic of computability (formalizability) theory is to demarcate the realm of systems and the realm of proteons and the limits of formal, systematic science and methods, as opposed to what I consider to be the central job of theoretical

---

[5] Many of the ideas presented in this section are adapted from the work of Kurt Ammon.

philosophy: the investigation of proteons. Its significance is therefore far greater than is current limited role inside theoretical computer science suggests.

2.3 Anomalies

When we encounter and describe a proteon, our description of it will always be incomplete in some way. This incompleteness will take several forms. Our everyday world is certainly a proteon: we will never be going to have a complete and exact description of it and what we know about it will always be incomplete. Therefore, we are familiar with most types of incompleteness from our everyday experience with the world. Some types of incompleteness of knowledge are the following:

- Gaps: you may describe some parts of an entity exactly, but some parts of the picture will be missing. Our "map" of the proteon will have white spots, so to speak.

- Idealizations (models, approximations): you may describe an entity exactly (as a system), but this description will only be an idealized or simplified approximation of the real thing. For example, an engineer's blueprint of a car seems to be an exact description of it, but it does not contain every detail of the microstructure of a real car and it does not describe all processes of wear and tear that finally move the real car out of the realm where that ideal description (the blueprint) is valid. The car will get scratches and rust and these processes are not covered by the blueprint theory of it. So the real car has more properties than the idealized model describes. The real car *emulates* the ideal one, but is not identical with it and eventually, the emulation is going to collapse.

- Faults and errors: the description might be partially wrong and might contain artifacts. An automatic car, for example, might "see" a street where there is none, and steer the car into a bumper as a result of this error in its knowledge.[6] The street "hallucinated" by the machine here is an artifact of the theory it contains.

- Inconsistencies: the description might contain inconsistencies and contradictions. We might be confused, but we might also, for some time, not notice these contradictions (or even push them out of our minds, e.g., for ideological reasons).

---

[6] Note that I am applying the concept of knowledge here also to knowledge stored inside machines and implemented in artificial systems. Further down, I will discuss some problems of what is claimed to be "artificial intelligence" in some more detail.

- Vagueness: you get only a blurred picture of the whole thing because some or all of the concepts you are using will be incompletely defined and might have to be changed and adapted in different cases of their use.[7] For example, if you look at the map of a city (an example of a non-conceptual representation of knowledge), red areas might denote buildings; but in some cases it might be hard to decide for the map maker if a given structure should be considered a building in this sense or not, and a clear formal criterion or algorithm to make this decision may not exist.
- Low resolution: in pictures (a special form of knowledge), resolution might be low or the picture might be blurred.

This list is probably not exhaustive. You might be able to come up with some other kinds of incompleteness.

There are cases where a complete description seems possible in principle, but is not feasible in practice due to its complexity. So instead of speaking of incompleteness, we might more generally speak of "anomalies" in our knowledge. A phenomenon might be an anomaly with respect to a given theory.[8] The types of incompleteness listed above are examples of anomalies but the concept of anomaly is slightly more general.

Sometimes, it is possible to reduce one kind of incompleteness or anomaly in a description by increasing a different type. For example, gaps might sometimes be hidden by introducing errors or inconsistencies. Gaps and errors might sometimes be "removed" by making the description vague. The other way around, you might be able to make a vague model or description more exact by making it more special, introducing gaps, or by introducing an idealization, removing some details from the picture, etc.

We may call such a transformation of a model or description by which the incompleteness or anomaly is "shifted" somewhere else, an "anomaly transformation."

In (Ammon 1987), Kurt Ammon presents a hypothesis saying that there is a trade-off between explicitness of descriptions and generality of descriptions. He

---

[7] There are attempts to understand the phenomenon of vagueness in terms of formal theory, e.g., fuzzy set theory. In my opinion, these attempts are unsuccessful. Vagueness, I think, arises when a concept does not have a fixed, stable definition. The definition can instead be adapted or reinterpreted with each use, and these processes of adaption can be creative processes themselves. The knowledge about what a word might mean can be thought of as forming a small analytical space (see section 2.4). In exact concepts, this knowledge is completely explicit and well defined.

[8] I have taken the term "anomaly" in this sense from (Ammon 1987).

refers here to what he calls “creative systems” (in my terminology, these are “cognitive proteons”, discussed below[9]), but I am supposing that Ammon’s hypothesis is valid for all types of proteons. The more explicit a description becomes the more special it becomes. If you try to make the description or theory more general, some parts of it will become vague or implicit or will have to be idealized away. There is, according to Ammon, a “forbidden zone” where we cannot go: our theories cannot become totally explicit and exact and at the same time totally general (if we are dealing with a proteon and not a system).

2.4 Analytical Spaces

Since every theory of a proteon is incomplete, our knowledge consists of bits and pieces. In (Ammon 1987, p. 73), Ammon introduces the concept of “Analytical Spaces”, as follows:

> Roughly speaking, analytical spaces form the components of creative systems. They consist of consistent but incomplete knowledge and the objects this knowledge refers to. The boundary regions of analytical spaces contain anomalies such as inconsistencies, inefficiencies, and gaps. Different analytical spaces are complementary in the sense that they cannot be unified into a single analytical space within a limited space of time.

A more concise definition is given on page 74:

> An analytical space of a creative system consists of a limited amount of consistent knowledge such as a few concepts and methods and the objects this knowledge refers to.

It is important to understand that here the knowledge and the objects it refers to are viewed together. The knowledge develops by its interaction with the objects it refers to. These objects have more properties than the knowledge covers, so the “creative system” (a human being or a group of humans, for example, or possibly an artificial entity), may generate new knowledge through interaction. Ammon gives an extended example that I want to cite here in its entirety (page 76):

> If a team of engineers has developed a new product such as a car or an airplane which is put on market, weak points of such a product are recognized after some time and the product is revised and improved. This process is repeated many times. The knowledge of the engineers about the

[9] I am not using the term “creative system.” This is because, in my terminology, such entities are not systems, but proteons. However, these are terminological differences only: Ammon and I are talking about the same thing.

> product and the product itself form analytical spaces. At any given point in time, the engineers have only incomplete knowledge about the product, which is revised and extended through experience. The product serves as the basis for the construction of new knowledge. The accumulated knowledge forms the basis for revising and improving the product. Thus, there is a coevolution of the knowledge and the product, i.e., the knowledge, the product and the interaction between the knowledge and the product run through an evolution process. This cyclic structure also applies to earlier stages of the product development process.

Unlike classical epistemological approaches, here the knowledge and the objects it refers to are treated as belonging together. The system starts with simple knowledge and evolves by interacting with the objects which, at any time, have some unknown properties.

On page 80, Ammon additionally defines the concepts of division and unification of analytical spaces:

> The generation of analytical spaces and the bifurcation of existing analytical spaces in creative systems is called division and the integration of different analytical spaces is called unification.… Cognitive structures evolve by division and unification processes of analytical spaces.

We can think of the analytical spaces as having a certain level of efficiency. During the development of knowledge, we might find methods and representations that allow to do certain things with higher efficiency, i.e., more economically and using less resources. So an anomaly consisting in low efficiency might be removed for certain cases (e.g. by finding a more efficient algorithm to do certain things). So generally, there is a trend towards increased economy of our knowledge. Ammon calls this the "Principle of Cognitive Economy" (p. 77).

3. Aspects of Proteons and Protean Aspects

3.0 Introduction

In the following sections I am going to look into protean aspects of different disciplines, in a rather unsystematic fashion. Along the way, additional properties of proteons are being investigated. Some sections are going to be technical a bit more demanding than others. The reader might skip these on a first reading.

3.1 Mathematics

I am not going to investigate proteons in mathematics or theoretical computer science here. Suffice it to say that starting at least with the work of Kurt Gödel, mathematicians have investigated the topic of non-formalizable entities in mathematics for almost a century. There is a great amount of literature about this, including textbooks, e.g., on computability theory, so there is no need to repeat this material here and I am just hinting at it.

3.2 Proteons in Physics[10]

There are several reasons for incompleteness or anomalies in physics, i.e., for the necessity to treat physical entities as protean. They might be open (i.e., being influenced by the environment), or they might be nonlinear in such a way that arbitrarily small changes in the initial condition are amplified to large effects (like in turbulence, e.g., in weather systems). There might be quantum mechanical indeterminism. There are also examples of physical processes that cannot be calculated in practice due to computational complexity—see, e.g., (Fraenkel 1993) on the protein folding problem. However, non-computability itself has to be added to this list as well.

Physicists are trying to come up with a complete theory of everything. As stated in the introduction, there seems to be a tacit assumption among many scientists that physics is computable. However, over the last decades several physical problems have been shown to be non-computable. A recent example is (Cubitt et al. 2015). The authors describe a proof for the undecidability of a particular physical problem, the "spectral gap" problem. In the introduction, they give an overview of work showing the non-computability of several physical problems. The authors have also recently published a description of their result for a general audience; see (Cubitt et al. 2018).

These results mean that physics is, in general, not computable, i.e. if physicists ever come up with a general theory of everything, this theory will be computationally incomplete.

We can expect this to be the norm, rather than the exception, for complex real world situations. Many simpler physical entities are probably still computable. For cases more complex (i.e., most real world entities), a proof of their non-computability might not be feasible due to their complexity, so in many cases, we

[10] Some parts of this section are based on (Keller 2017). At the time of writing that article, I was not yet aware that the non-computability of certain physical problems had already been demonstrated, see (Cubitt et al. 2015), so I presented this possibility as a hypothesis only.

might never know if they are proteons in the strict sense that a general formal description of them is impossible in principle. So there might be a rather narrow area of physical problems complex enough to exhibit non-computability, but simple enough for this to be provable. However, in practice, there are many cases in science where scientists do not proceed by solving the equations describing a "system" (or do so only for special cases) but use idealizations and approximations ("numerical models," "simulations"); instead, they replace the real entities with idealizations or models that are computable. The use of such approximations shows that in practice, physicists are familiar with the protean nature of nature, although some of them might not be self-consciously aware of it.

What does the statement, "that a certain physical system, or the structured set of entities modelling it, or the set of laws (in most cases: equations) describing it, is computable," mean? It means that we can solve the equations in all instances, using a finite body of mathematical knowledge. If we embody that mathematical knowledge in a computer program, then we could enter a description of the initial state of the system and calculate any later state of it at the push of a button.

A common way to describe this is what is known as the "DN" ("deductive-nomological") model introduced by philosophers Carl Hempel and Paul Oppenheim in (Hempel and Oppenheim 1948). According to this model, scientific explanations of a phenomenon (more specifically: a statement describing an observable phenomenon) E to be explained (the "explanandum") consists of an enumeration of a number of laws $L_1, L_2, \ldots L_k$ (I will be taking about the "*L*s" in my discussion below) and a number of initial conditions $C_1, C_2, \ldots C_r$ (the "Cs"). The explanandum is then logically derived (i.e. computed) from the initial conditions and the laws.

The common assumption that physical entities are computable then means that for any set of laws $L_1, L_2, \ldots L_k$ you can always derive E for any set of initial conditions $C_1, C_2, \ldots C_r$, using a finite amount of mathematical knowledge (i.e., a finite algorithm). This is normally taken for granted. It is, as I said above, a tacit assumption whose presence (and hypothetical nature) is not even realized by most people. But it is actually just that: a hypothesis. The non-computability results for some physical problems like the band gap problem mentioned above mean that it is actually wrong.

To be non-computable means that with any given set of mathematical methods to solve the equations, you can only cover a limited set of cases (although this set might be infinite). So non-computability does not mean that you cannot calculate any values of a non-computable function, but instead that any method to do so is incomplete. You may always add additional mathematical methods and thus extend the range of solutions, but the resulting theory (consisting of the physical laws and the mathematical methods to apply them) will always remain incomplete in the sense that you can set the initial conditions in such a way that the development of the "system" (or rather: proteon) cannot be calculated with the given set of methods.

In other words, we cannot take the derivation step, by which *E* follows from the *C*s and *L*s, for granted. It is possible (and has now been proven) that there are physical "systems" that are, in this way, computationally incomplete.

If the theory describing a physical entity is not computable, we would have to extend the DN-Scheme by another set of components. We would not only have the laws and the initial conditions, but also a number of calculation methods $M_1, M_2, \ldots M_q$ (the "*M*s"). To derive a particular explanandum *E* from the initial conditions and the laws, we would have to use a number of such computational methods.

In a computable system, there is, for a given set of model-structures, or a given set of laws, a finite set of such methods that covers all cases. We can therefore leave these methods implicit. This is done in the classical DN model, where the calculation is just hinted at by drawing a line under the *C*s and *L*s and it is implicitly assumed that we know how to compute *E* from the *C*s and *L*s. We can just say: "calculate" and assume that the physicist or mathematician (or computer) knows how to do that.

But if we don't know how to solve the equations, this does not work. The *L*s might describe the physical entity "completely" in an implicit way. But if we change the initial conditions, the *C*s, we might need to add some new *M*s to our list of mathematical methods in order to calculate or derive *E*. The physical theory, in such a case, is computationally incomplete. The *L*s by themselves are an implicit theory of the physical entity that cannot be made completely explicit. We see here computational incompleteness of "implicitly" complete theories as a new type of incompleteness.

The Laws (*L*s) and the mathematical knowledge (*M*s), together with the objects they are applicable to (represented by their descriptions in the form of the *C*s) can be considered as analytical spaces. The *C*s here represent the (descriptions of) the objects.

Note that the Popperian notion of falsifiability no longer works in such cases in its classic form. We can never be sure if all mathematical methods we are using (the *M*s) are correct. As a result we can only falsify the theory consisting of the laws (*L*s) and the computational knowledge (*M*s) together, not the laws alone. There could be errors in our calculations, while the laws are correct and we can never be sure our calculation methods are all correct[11]. Science therefore takes the form of a hermeneutic circle, in which we make sense of observations based on our current theories and methods and improve the theories and methods based on the observations.

3.3 Against Computational Metaphysics

Many people just assume that physics should be computable. Behind this is a concept of thinking of nature as a kind of a computer. According to this "computational metaphysics," nature works by somehow performing a calculation. That many people, in some quarters of Analytic philosophy, recently even seem to accept the idea that we might be living inside a computer simulation (Bostrom 2003) shows how widespread this confusion of physical processes and their calculation is. But a real physical entity just proceeds according to the physical laws describing it; it does not calculate them. In section 3.2, I have cited scientific results showing that actually some physical problems are non-computable. It is, therefore, no contradiction that some physical processes are not describable completely in terms of any single formal theory or algorithm. No matter how we set up an experiment, the physical entity ("system" in the parlance of today's physicist) would just develop according to the laws of physics. That we might not be able to calculate all cases with a finite set of mathematical methods does not stop the physical entity from behaving the way it does. The physical laws are a description of the entity, they are its invariants, but the system does not calculate.

So we should not assume that the possible is limited by the computable. It is, therefore, not necessary to assume, as some people, e.g., David Deutsch, have done (Deutsch 1985), that physics must be restricted to processes describable in terms of Turing machines (or Deutsch's extended version, quantum Turing machines). Computation is a process belonging to our models and descriptions; it does not exist in the real objects. It is a term belonging to epistemology, not ontology.

Likewise, nature contains processes that are computable in principle but whose computation is not possible in practice due to their computational complexity, e.g., the protein folding problem mentioned above.

One more point should be clarified here: Sometimes, the so-called Church-Turing-Thesis (or Church's Thesis) is wrongly applied to physics. This thesis states

[11] Roughly speaking, if we had a procedure to decide whether a given calculation method is correct, then we could enumerate all the correct methods by generating all possible programs from a given programming language and then deciding whether they are correct. We could therefore describe all correct methods in a finite way and add a finite method-generator to our theories. Any non-Turing-computable theory would then become Turing-computable. So assuming that such a procedure exists leads to a contradiction.

that all *formalisms* describing the concept of computability (or formalizability) describe the same set of functions. This is not a thesis about the physical world, stating that physics must be computable and the computable equals the possible, but it is a statement about formal systems (and an unproven one). A physical entity is, however, not a formal system. The Church-Turing-Thesis might be true. But physical reality can still contain proteons, i.e., objects that cannot be described completely in terms of a single formal theory or an algorithm.

3.4 Proteons in Biology

Organisms and higher order entities like ecosystems and the biosphere are, I am supposing, proteons. However, within the limited space of this essay, I am going to discuss only a few aspects of the protean side of biological entities.

In particular, biological evolution is a protean process. In an organism, there are some processes that are under the control of the genes. These processes represent the normal functioning of the organism. But there are always some processes that are not under the control of the genes.

For example, if you look at what is going on in a cell, e.g., in a single-celled organism, you will notice that some of the processes happening there, like some of the movements of organelles in the cell's plasma, are clearly random. They are not under the direct control of the cell's genes. Likewise, the molecules in the cells move and react in ways not dependent on the control of the genes. The genes just constrain these processes to some extent.

In particular, mutations of the organism's genes themselves are not under the genome's control. There might be some controlled processes of gene editing, like crossing over between chromosomes etc. But generally, the processes changing the genes are not controlled. And this is important. If all mutations were completely controlled, the evolutionary potential of the organism would be limited to information already contained within its genome. An organism does not contain enough information to "know" how to evolve into another form (that might be more complex, although this is not always the case). A modification from the outside or by a random process, the mutation, is required because of this insufficient information. There is a parallel here with the inability of algorithms to evolve, to increase their own complexity. If the genes would control evolution, all "evolution" in such an organism would actually be an instance of ontogenesis, as opposed to phylogenesis, and the organism could actually not develop into anything new. It would not be evolving in the sense the word is normally understood.[12]

---

[12] In the 19th century and during the first half of the 20th century, there have been some "orthogenetic" theories of evolution, mostly in connection with vitalistic concepts (I want to

To use a concept from computer science here, we could say that an organism is "emulated" by a physical system. Some of the properties of the physical entity "running" the organism are under the control of the organism's genome, while others are not. The description of the organism as a gene-controlled system is incomplete with respect to the physical entity emulating it. The physical system has more properties than the genome describes. There is a "residuum" of properties or processes outside of the control of the genome. Moreover, the organism is in constant exchange with its surroundings. Mutations act out of this residual part of the entity (e.g., its quantum-mechanical microstructure) and out of the environment.

The controlled part of an organism might be described in terms of a formal theory (whole-cell simulations exist already for simple organism, like some bacteria, and scientists are working on simulations of some simple animals, like nematodes). But such models are incomplete, especially in the sense that they fail to describe the potential of the organism to evolve. And while some processes of evolution may be modelled inside a computer, true, open-ended evolution cannot be described in terms of an algorithm alone.

The bilaterian animals to which we belong developed from simpler animals like sponges that in turn developed from single-celled organisms (a kind of flagellate). And these developed out of even simpler organisms. A complete description of these simple, single celled ancestors would have to include all the organisms they could evolve into, including human beings and all of their culture. This is clearly impossible.

3.5 Science as a Proteon

If reality as a whole and many or even most parts of it, including reality as described by physics, is a proteon, then science has to be a proteon itself. If the scientific method could be described completely in terms of a single formal theory or algorithm, science would necessarily be incomplete and blind to some aspects of reality. Instead, in order to be universal, the scientific process has to include the possibility of going beyond any fixed set of methods and algorithms, concepts, and representations. Science must be able to adapt and change any aspect of its structure. It has done so in the past, enabling it to adapt to varying circumstances reality imposed on it and it must retain this flexibility in the future. As a result, while there

---

emphasize here that my view that life is protean is not a vitalistic view, but this topic is outside the scope of this article). Such concepts would require that evolution is not a random process but somehow guided. If this were so, organisms would have to contain the information about their future evolution. If we apply the parallel with formal theories or algorithms the other way around, we could view algorithm-based theories of learning and AI (discussed below) as "orthogenetic," sharing a common mistake with orthogenetic theories of biological evolution.

can be a philosophy of science, there cannot be a complete science of science, at least as long as "science" is viewed as the process of producing descriptions of reality in the form of formal theories or algorithms. Science a whole must then be outside of its own scope.

However, we might replace this (unnecessarily) narrow view of science with a more comprehensive one that includes the topic of proteons and admits that complete formal theories are not always possible. The result would look more like what is meant by the much wider German term "Wissenschaft," which has always comprehended disciplines excluded from science in the narrow sense, like "Kulturwissenschaft," "Geschichtswissenschaft," etc.

When the concept of science is extended in this way, the division between the "two cultures" (Snow 2001, 1959) dissolves. This division can be seen as an artificial result of restricting science to formalizable entities. By imposing this restriction, the academic disciplines are artificially split into the "sciences" and the "humanities" and subsequently, different groups of people gather around these two sections of knowledge, forming the two cultures observed by Snow.

The separation is overcome not by a kind of reductionism in which the more complex "humanities"—which have always been dealing with the full complexity of proteons and have developed methods to do so—are reduced to scientific objects that are fully formalizable. Instead, the separation of the two cultures can be overcome by science consciously opening up itself to protean aspects of reality. As a result, fields of study like social sciences or psychology that have long been trying to establish themselves as proper sciences, for example by making heavy use of statistical methods in order to bring numbers into the field, can open up their methodological repertoires. The rift between the two cultures goes right through the middle of these fields, and is the artificial result of a narrow concept of what science should be, hence of a narrow view that concentrates on systems, i.e., non-protean aspects of reality.

3.6 Cognition as a Proteon

If physics contains proteons, then reality is a proteon. What about human beings? What about our cognition?

There is a tendency in science to try to build formal theories of everything. In the field of disciplines whose subject is the human being, this takes the form of a cognitive psychology that is oriented towards the fields of "artificial intelligence" and robotics inside computer science. This line of research is also closely connected to certain areas of Analytic philosophy and to formal linguistics. In a nutshell, it is the attempt to describe human cognitive processes (perception, thinking, action, and language processing) in terms of algorithms, i.e., formal theories.

As I am going to show below in section 6, every algorithm is limited in its capabilities and has systematic gaps or "blind spots." More specifically, I am going to sketch a proof of this in section 6 of this essay.[13] The reason I am shifting this to a separate section at the end of the essay is that I don't want to overburden the reader at this point with technical details. In a nutshell, what is shown there is that algorithmic artificial intelligence is impossible, since every "AI" system that is limited to algorithms, has gaps. This also holds for learning algorithms: for every learning algorithm there are problems (perhaps learnable by some other learning algorithm) that this particular algorithm cannot master.

If human cognition could be modelled or described completely in terms of an algorithm, i.e., if humans are systems, this would mean that there could be aspects of reality which humans could never understand, due to the limitations of the architecture of their cognitive apparatus, even if other algorithms capable of understanding these particular aspects of reality are possible. I have discussed this with respect to science in section 3.5: there could be aspects of reality that would be systematically out of reach of human science because of the way our brains work and are programmed. We would be in a situation of a fly repeatedly bumping into a window pane. We could not understand the difference between transparent glass and air. We would be systematically blind to certain aspects of reality.[14]

If, on the other hand, we assume that our ability to understand reality is universal in the sense that there are no such blind spots that we could not overcome, then our cognition cannot be an algorithm. We would have to be proteons ourselves, just as our science would have to be a proteon, not a system.

[13] A more formal presentation of the proof can be found in (Ammon 2013).

[14] It appears to me that Kant's system of categories and forms of intuition imply a fixed structure of cognition that he thought to be a necessary condition for the possibility of the acquisition of knowledge. This fixed structure leads to a separation of reality into a world of phenomena reachable for our perception and cognition and a thing-in-itself that is unreachable in principle (and it looks like he wants to have that impenetrable wall there to hide things like God behind it). In contrast to this, I consider the categories and forms of intuition (or any other comparable cognitive structures) to be subject to change through learning processes. Such structures can arise from simpler structures (which in turn can arise from even simpler ones), so the limit of the inaccessible thing in itself can be shifted (the inaccessible is the part of the protean reality not yet covered by our formal descriptions at a given time). We might start with some innate structures since starting from nearly nothing (i.e. from cognitive structures of very low efficiency) takes too long, but these innate structures could also arise in a learning process, just as they have arisen in the process of evolution, and they are not a fixed structure of the mind but just the point of departure of our learning processes. Moreover, it seems to me that our biologically innate structures of cognition are simpler than Kant's categories and that these are, at least partially, a product of our culture.

We would be cognitive proteons (or "creative systems" in Ammon's terminology). Perhaps parts of our cognition and our knowledge can be described as algorithms and formal theories, but we would need to have the ability to move or develop out of the scope of any formal theory or algorithm modelling or describing us. The AI-paradigm of cognitive psychology would be wrong. A *systematic* description of the structure of human cognition would be impossible, because there could not be such a fixed structure. The "laws of thinking" would turn out to be changeable, re-programmable so to speak, and historically developing. They could be different in different individuals, groups or institutions, in different cultures and sub-cultures and at different times in the development of an individual and of a culture or society.

If we assume that human cognition is a process of the brain or of the whole living body (i.e.. if we assume a non-dualistic position), and our brains/living bodies are physical entities, (on a neurological, cellular or chemical level), then our brains/living bodies are proteons in the sense of physical proteons described above. In the section about proteons in physics, section 3.2, we have already seen that physical proteons actually do exist.

In this essay, I am indeed pursuing that very possibility: human beings are proteons, hence human cultures and institutions, with their histories, are proteons as well. The process of science is a proteon and human cognition is universal and does not have a fixed structure. A defining property of humans is their creativity, and creativity can be defined (as has been done by Ammon) as just that ability to move out of the scope of any existing theory, exact model, or exact description.

This does not necessarily mean that artificial intelligence is impossible. But what is currently being presented under that title does not deserve it. Current "AI" systems are just that: systems. They are algorithms, each having certain blind spots. The real problem of what it means to create an artificial intelligence has not even been seen by the AI researchers yet. The real problem of AI is how to build an artificial proteon, i.e., an entity that can change every aspect of its structure, an entity that is historically developing and changing and that cannot be understood completely in any systematic way since every formal theory about how it works must be incomplete. The research paradigm of AI was created in the 1950s with the goal of producing a systematic understanding of cognition (see Ammon 2016, discussing McCarthy's 1955 proposal for a 1956 workshop on AI). The researchers involved did not understand that what they were trying to do was self-contradictory.

3.7 Language and Linguistics

The attempt to turn linguistics into a "science" in the narrow sense of the term has resulted in the development of formal linguistics by people like Noam

Chomsky.[15] Chomsky deals with the complexity of language by an idealization. He concentrates on the "grammaticality" of sentences, thus carving out a formalizable area out of the total reality of language. What he is interested in is what he calls "competence," the ability to distinguish grammatical or "well-formed" sentences from non-grammatical ones. The use of language with all its complexities (what Chomsky calls "performance") is defined as being outside the scope of linguistics. Chomsky made tremendous contributions to the development of the theory of formal languages and thus to the theory of computability and to theoretical computer science. He also made important contributions to the development of grammar, helping to free it from the restrictions of traditional grammar. However, by the restriction to what is formalizable, the topic of linguistics is turned away from actual language as used by actual people.

Chomskyan linguistics can describe language only as a fixed code. It is therefore blind to the processes of language acquisition (for which Chomsky envisages a kind of an algorithm, a "language acquisition device" that nobody ever could describe in detail), to the processes of language change, to extensions of language and to the historical origin of language.

Speaking and understanding speech do not require that the "code" of the language used is completely defined in every detail. The listener (or reader) can still understand an utterance even without having complete knowledge of the exact meaning of all the words and of the exact details of the grammar. We are normally able to understand an utterance spoken in a dialect that is slightly different from our own and are often able to understand utterances or written messages expressed in closely related languages. When learning a foreign language, we are often able to find out the meaning of a word new to us from the (linguistic and situational) context. We can also cope with mistakes and noise. We can cope with mistakes made by a non-native speaker or a child. Applying information from the context and from the situation, we can correct the errors and fill in the gaps. Understanding is not just the "mechanical" or "algorithmic" application of a code or a grammar, it is a creative activity that does not require the utterances we hear or read to be completely well-formed.

Moreover, for our language to work, it is not necessary for every word to have a complete and exact definition. And in fact, most words are more or less vague. Their definitions are incomplete and are modified by speaker and listener on the spot all the time, leading to redefinitions and shifts of meaning.

---

[15] I am taking Chomsky here only as a paradigmatic example. I am aware of the fact that there is much more to linguistics and language oriented AI than just the works of Chomsky, but he is one of the leading proponents of this paradigm and the problems can be exemplified with his work.

Vagueness of utterances and of terms means that the use of language is not and cannot be completely formalized. It is a creative process for which there is no complete model or description in terms of a formal theory, a formal grammar, or a set of algorithms. Language is a proteon. This incompleteness entails the possibility of misunderstandings that is an unavoidable feature of language, but we should not regard this incompleteness of the language's system and the vagueness of words and utterances as a defect of language. Instead, it is an intrinsic part of language important for its proper functioning, adaptability and flexibility. Language use is always embedded in a situation, and language as a system is embedded in general cognition. The totality of cognitive processes can contribute to its working and this totality cannot be completely described in terms of formal systems.

When a child learns a language, it starts without any knowledge of it. But the intrinsic creativity of language use is what makes the process of language acquisition possible. The child can use words and syntactic constructions without knowing every detail in advance, and its utterances can be understood even if they are faulty with respect to the "complete" system of the language (if anything like a "complete" system even exists).

The ability of the listener to make sense of an utterance results in a certain tolerance for changes. The language of the child will work even if the child fails to completely reproduce the system of the language as used by the previous generation. Innovations (with respect to the "system" as used by the previous generation) might then be adopted by other children. As a result, languages change.

The creative process of language acquisition even works when there is no completely defined target language. In situations where people with different languages come into close contact, they can invent simple pidgin languages. The next generation of children, growing up in an environment where such a pidgin is spoken, can "negotiate" among each other the rules of a new emerging language, resulting in a creole language that is as complete and well-equipped as any other language. This process of "negotiation" is, in most cases just happening through the creative invention and creative understanding of new bits of grammar or lexical code.

And this creative side of language use is ultimately what made the emergence of human language possible in the first place. The process of communication does not require the pre-existence of a complete code or system of grammar and words. A more complex system can arise from a simpler one, i.e., a language can be extended since speakers can invent new words or meanings and new bits of grammar, and even new kinds of meanings, new semantic or logical "devices," e.g., the use of universal quantifiers.[16] Listeners can understand what is meant by making sense of it, using information from the situation. And this process of developing more complex and elaborate languages from simpler ones may ultimately start with an empty system.

So at some point in history, "natural" language was invented (and since it was invented, I think the term "natural" language often used in linguistic contexts is actually a misnomer). All that was necessary for this to happen was that general intelligence surpassed a certain threshold. I don't see the necessity of any specific language ability to arise biologically first (although there might be secondary genetic adaptions to it later).

To sum it up: we are using language, but our knowledge of each language we are using might always be incomplete. This incompleteness might show itself in the form of gaps in the system and in the form of vagueness. Communication still works because it is a process involving creativity on all sides. The "complete system" of a language, describable in terms of formal grammars or algorithms, might not even exist since languages are constantly changing through these creative processes.

3.8 The Cognitive Landscape

Moving out of the scope of a theory, we are moving from one formal theory to a more comprehensive one. We might call such a move a "creative step." Ammon also speaks of "structure breaks" (Ammon 1987, p. 82). Obviously, there is a connection to Thomas Kuhn's work (Kuhn 1996). There is also a connection to the "Ebenen und Krisen" or "Plateaus und Krisen" that Carl Friedrich v. Weizsäcker writes about in (Weizsäcker 1977). It is likely that other thinkers have observed such patterns of development in philosophy, in psychology, in pedagogy etc. For example, Ammon notes a connection of his work to the psychology of Jean Piaget (Ammon 1987).

The paradigms of Kuhn and the plateaus of Weizsäcker correspond to Ammon's analytical spaces. Cognition can be thought of as a process in which information is assimilated into analytical spaces until anomalies force new analytical spaces to be generated or existing ones to be divided. At times, several analytical spaces can be unified into larger ones (a prominent example is Newton's the unification of different terrestrial and celestial mechanics into a common

[16] What is known as the "semiotic revolution," the apparent emergence of more complex cognition at a certain time during the stone age, might have been triggered by such a—completely cultural—extension of language, something like the invention of quantifiers. There is no proof that it was based on a genetic change. Daniel L. Everett's work on the Piraha language shows that languages without universal quantifiers and numbers are perfectly possible, demonstrating that these are features of culture, not genetics; see (Everett 2005).

framework). In reality, the development of our cognition might be much more fluid. The creative processes might be happening all of the time and it might not be possible sometimes to discern clearly defined "levels" or "plateaus" separated by more or less instable phases of change. We might think of such processes also in terms of the development of analytical spaces, but quite small ones.

Metaphorically, we may think of a cognitive landscape through which we are moving, with some flat plateaus (the large analytical spaces of the classical sciences), some steep faces or slopes where we have to climb from one plateau to another, and some areas with rugged terrain where we cannot really make out any flat areas.

3.9 Pedagogy, Social Pedagogy, and Therapy

> *Truth and reality in art begin at the point where the artist ceases to understand what he is doing and capable of doing…*
>
> —Henri Matisse

If human learning develops as a sequence of "plateaus" that can be viewed as analytical spaces, our understanding at each stage of our development is limited to what can be understood and thought in terms of the analytical spaces we inhabit at the given time. We can develop further by "climbing" to "higher" plateaus, i.e., developing more comprehensive cognitive structures. However, the previous stage of development does not contain enough information to clearly lead us to the advanced stage. We might get there with luck, but normally it helps to have a teacher or mentor. This also holds for processes of therapy, developing out of states that might be thought of as pathologic in a psychological or social sense.

The way leading to the advanced stage can only be understood in hindsight, from the point of view of the more advanced knowledge (just like the gap in a mathematical theory can only be filled inside an extended, more advanced theory). What is happening here is not that an unchanged person with a constant human nature is "filled" with some skills and knowledge, like a container is filled. Instead, the human being is *transformed*. There is no fixed structure of cognition and any aspect of cognition can be transformed.

So learning is about transformation of ourselves. It is about becoming who and what we can become. We should therefore view access to education, to learning, to "Bildung" (a German term referring to education that can be literally translated as "(trans)formation"), as a basic human right. Each human has a basic right to get as far as possible in the development and advancement of knowledge and personality. "Bildung" should therefore not be treated as some commodity to be paid for. It should be available for free to anybody. Since the individuals who work as educators, teachers, mentors, tutors, counselors, social workers and therapists must be paid, societies must pay them to make sure anybody who wants has access to educational and therapeutic services of all kinds.

3.10 Theoretical Philosophy

> *Grau, teurer Freund, ist alle Theorie*
> *und grün des Lebens goldner Baum.*
>
> —Johann Wolfgang von Goethe, Faust 1[17]

Our science and scholarship deals with reality. If reality is a proteon, we must expect that each of our scientific and scholarly disciplines can only cover certain analytical spaces. Some of these are very large (like in physics) and moving around in such large analytic spaces, we will not see the horizon and might come to the conclusion that they cover everything, but this is an error. If we look at the process of human knowledge acquisition through history as a whole, we can see that new sciences form whenever an analytical space becomes very large and comprehensive. The areas remaining, where analytical spaces are small and problems are difficult, form what is known as (theoretical) philosophy. Whenever an analytical space becomes a science, it stops being part of philosophy. One gets the impression that there is no progress in philosophy. But exactly that sprouting of new sciences from the main stream of philosophy is an essential part of its progress. However, the unification of all analytic spaces into a single one is impossible. Necessarily, philosophy remains somewhat vague. It remains divided into several traditions, and attempts to systematize everything into one big systems repeatedly fail. Such projects must fail if reality is a proteon. As a result, we should be skeptical about any claim of such big unifications, which often turn into ideologies, i.e., pseudo-complete theories of an idealized reality. "Systematic philosophy" is an oxymoron since philosophy is basically about proteons.

If philosophers try to turn theoretical philosophy into a science (in the narrow sense), i.e., to make it systematic and formal—and I think that this is basically the project of at least one brand of classical Analytic philosophy—the resulting descriptions necessarily become special. Generality and exactness cannot be achieved at the same time. The result is the increasing specialization that can be observed in Analytic philosophy. The attempt to formalize everything causes philosophy to disintegrate into small special analytical spaces many of which are of little relevance for any important question. Since formalization is easiest in the area of science (in the narrow sense) and fails in the more complex realms of the humanities, Analytic philosophers concentrate on the former. However, the

[17] Translation: "Gray, dear friend, is all theory, and green the golden tree of life."

scientists, especially the physicists, inhabit large analytic spaces where philosophical thinking (climbing to another plateau of the cognitive landscape) is not required most of the time. So this is the area that has the least need for philosophy. As a result, Analytic philosophy has made itself dispensable and irrelevant. There is a great need for philosophical thinking in the more protean areas of psychology and cultural and social disciplines, but Analytic philosophy has nothing to offer there. The practitioners in those areas develop their own philosophies because the professional academic philosophers are not producing much of use for them. Likewise, for those theoretical situations in which a paradigm shift is needed in the sciences, e.g., when quantum mechanics was developed in the 1920s, the physicists involved developed their own type of philosophy, and this philosophy was quite distinct from the theories that the precursors of Analytic philosophy (the Vienna circle, etc.) had to offer. Generally, I see a need for a new, proteon-oriented or "creativistic" paradigm of philosophy to replace the Analytic philosophy that has dominated professional academic philosophy since the end of World War II, i.e., for the last 75 years.

If theoretical philosophy is about proteons, this has consequences for the form of philosophy as well. If it is not about formalizable systems, then there is no reason to restrict the language of philosophical thought to the kind of systematic papers to be found in the analytic journals. Instead, philosophical thinking should be free to use any form, including poetry and even non-textual forms, since we cannot know in advance what form of thought will get us where we want to go. We might try to systematize things in hindsight, but as long as we are in exploratory mode, everything must be allowed. The kind of equipment that is appropriate to navigate and move around on the large, flat plateaus of scientific analytic spaces might not be useful at all in the steep and rugged terrain the philosopher is climbing. A path to a new location might first be discovered through a poetic form, and only later it might be possible to build a formal ladder or staircase.

3.11 History of Philosophy

Within the history of philosophy, we might distinguish between a current of thought that we could call "essentialist," starting with Plato and ending in contemporary Analytic philosophy, and on the other hand another current that denies the existence of fixed essences, forms or structures, and views reality as something more fluid. One could try calling this "existentialist," if that term was not already covered by something more specific. In any case, this non-essentialist, "proteonistic" tradition would start with the Sophists and perhaps Heraclitus. Certain forms of historicism, empiricism, and Existentialism can be put in this current. Indeed, we could go through the history of philosophy and sort thinkers into those who, on the one hand, tried to build systems or who believed that an essence of reality that could be described completely existed, and those who, on the other hand, thought this was impossible. Note that this distinction is *not* identical to the distinction between Analytic and "Continental" philosophy—a distinction I consider to be useless anyway. For example, Husserl, with his "Wesensschau" certainly belongs in the "essentialist" camp, while other phenomenologists—e.g., Wilhelm Schapp with his phenomenology of stories, see (Marquard 2004)m—clearly belong on the non-essentialist, "proteonistic" camp. Other examples of philosophical traditions belonging mainly on the non-essentialist, proteonistic camp are philosophical hermeneutics, and philosophical anthropology with its notion of the "world-openness" of the human being—see (Fischer 2009).

3.12 Practical Philosophy, Ethics, and Law

> 道可道，非常道。名可名，非常名。
> 道德經[18]

I want to provide only a few hints here about the topics of ethic and law. A comprehensive treatment doing justice to these topics would require by far more space.

Our social reality is a proteon. It is, therefore, a mistake to try to formalize law and ethics in the way of systems. A partial formalization might be possible and necessary, but it must always be possible to go beyond the systematic framework in order to adapt it to individual situations.

We should meet attempts to build formal, systematic philosophies of ethics with skepticism. Any such system is bound to be incomplete and is thus going to lead to un-ethical results in some cases. Systems are important but we must operate above them. Ethics must be open and creative to cope with the complexities of a developing world consisting of creative humans.

I see a connection here to the work of David Chapman[19], especially his concepts of meaning being both patterned and nebulous, and his notion of meta-rationality.

---

[18] This is the first line of the Dao De Jing; see the "Chinese Text Project" (2006-2019), available online at URL = <https://ctext.org/dao-de-jing/ens?filter=625839>. The translation by James Legge given there is: "The Dao that can be trodden is not the enduring and unchanging Dao. The name that can be named is not the enduring and unchanging name." Note that the term "Dao" that Legge leaves untranslated here originally simply means "way."

[19] See D. Chapman's hypertext book project, "Meaningness," available online at URL= <https://meaningness.com/>.

It is necessary to warn against any attempts to introduce algorithms as judges or in similar areas.[20] Any attempt to replace a judge with an algorithm is going to lead to injustice. A judge is an intelligent proteon able to go beyond any special systematic or algorithmic "judgement" of a situation, where this is necessary. Any algorithm, on the other hand, has systematic blind spots which will in some cases lead to unjust results.

Similarly, attempts to formalize work processes, inside private companies or public administration, are going to lead us into problems. Systems are necessary and helpful and bureaucracy is unavoidable beyond a certain level of complexity, but we must remain flexible and operate above the systems, with people retaining the right to make decisions, or else we will run into severe problems.

3.13 Aesthetics

> *Im Übrigen zeigen heute viele, wie gefährlich es ist, wenn eine „Idee" zum „System" führt, zur „Formel", die stets den Todeskeim in sich trägt. Denn nur im Augenblick ihrer Entstehung, ihrer Erfindung ist eine Idee lebendig.*

—Eduard Bargheer in a letter, November 10th, 1949 (on exhibition in Eduard Bargheer Museum, Hamburg, Germany)[21]

I am also only hinting at possibilities I see opening up in the area of aesthetics. On the one hand, we can try to describe the process by which works of art are created in terms of the concept of creativity introduced above. On the other hand, we can also look at the process of perception of art or music or other things.

Philosophical aesthetics is relatively helpless in the face of the phenomenon of beauty and has largely defined it out of its subject area. I suppose the reason is that this phenomenon is tied to the protean aspects of the process of perception. It cannot be understood inside an essentialistic, formalistic, or analytic framework. Knowledge is used to interpret the stream of raw sense data. This knowledge forms analytical spaces. It is used to form expectations and to integrate sense data along the lines of regularity contained in it. If we are unable to find any regularity, the

---

[20] See, e.g., H. Hodson, "AI Gets Involved With the Law," *New Scientist* (15 May 2013), available online at URL = <https://www.newscientist.com/article/mg21829175-900-ai-gets-involved-with-the-law/>.

[21] The recipient of this interesting letter, written by the painter Eduard Bargheer, seems to have been Bargheer's friend Susanne Bonte. He is referring to painters. And here is a translation:

> Moreover, many [painters] today show how dangerous it is when an "idea" leads to a "system," to a "formula that always contains the seed of death. For only at the moment of its creation, its invention, is an idea alive.

result is confusion. If all the sense data can be assimilated into existing analytical spaces, i.e. if it is as expected and free of surprises, the result is boredom. Between these two extremes is an area where there is some surprise but on a level we can cope with: we can integrate the new information by extending analytical spaces. My hypothesis is that the experience of beauty is a sense of achievement (*Erfolgserlebnis*) that occurs when a new (surprising) piece of information is successfully integrated into an analytical space. In this process, the knowledge is expanded, i.e., a cognitive change takes place. Since different individuals and individuals with different cultural backgrounds differ in their perceptual knowledge, the valuation of things as beautiful is very subjective and culture-dependent. It can also change over time: if we experience the same thing (e.g., a piece of music or an abstract painting) over and over again and get to know it, it will not produce the same experience of beauty again and might even turn boring. One result of this is that artistic and musical styles often have a tendency to become more complex over time.

3.14 The Human Condition

There is a crack in everything, that's how the light gets in

—Leonhard Cohen

Life is protean. Reality is a proteon. Humans beings are proteons as well. As a result, existence precedes essence (not just for the human being, as Sartre thought, but for everything except those facets of reality that can be described as systems. Reality is a proteon and we have to face it as proteons.

As individuals and as societies, we are biographically and historically changing. There is no fixed human nature. What we are when we are born is not our fixed structure, but just our point of departure.

The basic and, in my opinion, defining property of the human mind is creativity as defined by Ammon: the ability to break out of any scheme, any fixed pattern or law of thinking.

This results in our world being "open," in contrast to the closed, defined worlds of non-human animals. We are able to grasp new phenomena and re-interpret known ones in new ways and in doing so, we are able to extend our world and our understanding of it as well as enter into novel interactions with the world.

A single unified and unifying theory of human thinking and human culture is, as a consequence, not possible since we are able to break out of the bounds of any such theory. It is therefore impossible to come up with a complete description of the human mind and of human cultures. We cannot understand or define ourselves completely. This is reflected in the vast plurality of topics and methods of the humanities.

As a result of creativity, our cultures split up into an unbounded multiplicity of different groups showing an unlimited multiplicity of different phenomena. One way to deal with this situation is to embrace it and accept the multi-cultural and pluralistic global society resulting from it as something positive. This way of reacting to our fundamental "world-openness" leads to the development of the ideals of tolerance and a pluralistic,democratic, and cosmopolitan open society.

Some people, however, seem to experience the fundamental incompleteness of our self-definition as a threatening gap, as an abyss. They try to patch this gap in different ways, by means of institutions limiting the freedom of action and by means of dogmas limiting the freedom of thought. Such systems may come in the form of religions or any kind of political ideology. The "gap" can always be ripped open again by creativity, so creativity is a threat and must be limited. Limiting the freedom of thought therefore results in the disruption and crippling of creativity. Creativity, however, is at the core of intelligence, so attempting to close the "gap" for good results in humans and societies that have reduced their own human and cognitive potential. One could say: it is not stupidity that results in ideology, but ideology that results in stupidity.

As long as ideological thinking is retained in the private sphere, however, it hardly causes problems for an open society. The attempt to build a society based on any such ideology, however, will almost necessarily lead to violence. The creativity of humans can always break out of the limitations imposed by a religion or ideology. Since this would undermine the basis of a society based on the religion or ideology, creative thought as well as the people who think or behave differently must be suppressed, in one form or another. This is only possible with some form of violence, from brainwashing and propaganda, through peer pressure and suppression of books and free speech, to imprisonment, torture and killing. So the attempt to create a secure and fixed foundation of society and morality causes immoral results. Similarly, such societies and cultures will tend to get into antagonism against other societies and cultures, resulting in conflict up to the level of open war.

The attempt to turn the open, creative human being into a closed follower of any fixed system of thinking can thus be seen as one of the main sources of evil. To avoid it, I can see no alternative to an open, pluralistic, cosmopolitan, and tolerant society. However, not all cultures can participate in such a multi-cultural society. People or groups who are not able to apply a minimum of tolerance themselves and who claim to own absolute truth must be excluded.

The challenge lies in doing this in a way that does not compromise the basic openness of society. This might lead into dilemmas and difficult problems in many cases, but the basic values of openness and freedom must be kept up or else our societies will turn themselves into some form of closed ideology, be it patriotism or whatever. Solving these problems can be very hard, and limiting creativity here causes deadlock situations in which opposing ideologies are facing each other. We must not succumb to the temptation to patch the gap. Doing so is, in my opinion, actually a childish attitude.

Instead, we have to bear the fundamental and unavoidable incompleteness and patchiness of our own human existence. The inability to give complete answers and find perfect solutions is the inescapable back side of that same creativity that makes us what we are: human beings. We should understand this as a source of strength, and we have to use this same creativity to try to tackle the problems resulting from it.

4. AI and the Limits of Formalization

4.0 Introduction

In this section, I am going to present several arguments showing that algorithms are limited. Some (although not all) of these arguments are a bit more mathematical than the previous sections, and that is why I have put them into this separate section.

One line of thought starts with the idea that learning can be viewed as information compression. Each algorithm can be viewed as the representation of only certain patterns. A data stream longer than a given algorithm that can be produced or parsed by that algorithm must necessarily contain some specific regularity. Additional information outside that specific pattern cannot be produced or parsed by the particular algorithm. It has a systematic blind spot.

Second, the non-computability of the Kolmogorov complexity means that no algorithm can find every instance of regularity in arbitrary data.

Third, I am going to introduce the notion of total functions as a mathematical model for knowledge and present a sketch for a proof that computable total functions are not Turing-enumerable.

4.1 Defining Intelligence

It might help here to try to define the concept of intelligence.

In order to deal with the protean nature of reality, our cognition (perception and thought) has to be a proteon itself. Likewise, the collective activities we denote as sciences and scholarship must be protean as well.

As noted above, I think that all formal theories and algorithms are limited. Each one can only produce or process a limited set of patterns. Each one has systematic blind spots. Universal intelligence, on the other hand, requires creativity, i.e. the ability to step out of the framework of each limited formal theory. Universal intelligence can overcome the blind spots of each formalizable single piece of knowledge. It is therefore not formalizable in its entirety. It is protean. A creative cognitive entity (a cognitive proteon) has the ability to move out of the scope of any formal theory used to describe it. If such a theory is extended to include a process of moving out of the scope of the previous, more limited theory, the new, more comprehensive theory will turn out to be incomplete again.

We can therefore define intelligence this way: a cognitive entity is intelligent if and only if (i) it has formal reason, i.e., the ability to apply formal knowledge, make logical inferences or algorithmic calculations, and (ii) it is creative, i.e., able to step out of each single piece of knowledge, structure or process inside it that can be modelled by formal theories or algorithms, and generate new ones. In short:

> Universal (protean) intelligence consists of logical reason (ratiocination, calculation, logical inference) plus creativity.

Intelligence, defined this way, cannot be achieved by any single algorithm. Current "AI" systems are not intelligent in this way since they are missing creativity and can be described as algorithms (more on that below).

4.2 Learning as Information Compression

One way to see the limits of formal theories (or algorithms) emerges from looking at cognition as a process of information compression. This is not a new idea, but I don't want to go into the literature about this idea here. In a nutshell, in cognition we learn something about reality. Our knowledge allows us to make predictions, so some things will no longer surprise us. Thus, the knowledge enables us to make use of the regularity of the phenomena. As a result, it can be viewed as a compressed form of the data streaming into us through our senses. Where our expectations do not fit that data stream (that is where we encounter surprises or anomalies), we learn something new.

To make the limitations of algorithms in this framework understandable, I want to compare this process to something much simpler, something that at first does not look like an algorithm at all: cylinder seals. I hope that this will also clarify some points made before.

Cylinder seals were invented in Mesopotamia around 3500 BC. They are small cylinders, often made from stone, about one inch in length, engraved with figurative scenes or written characters, or both. They were used to roll an impression on a surface, typically clay.

When the seal is rolled over a clay surface, an impression is formed. If the seal is going through more than one full rotation, the image printed into the clay will show a partial or complete repetition. As soon as the printed image is longer than the circumference of the seal, there must be repetition.[22]

Repetition is a simple instance of order. If an image, or more generally a piece of data, contains some order, it must be possible to compress that data into a more compact form. And indeed, the cylinder seal may be viewed as a compressed form of any image produced by it that is longer than its circumference. If the seal can generate the picture and the picture contains a higher amount of data than the seal itself, the seal can be viewed as a compressed form of that data and the generated data must contain order or repetition. The only information added on top is the information of how many times the seal was turned and at which point of the seal the image started.

We can view the seal as a simple special-purpose printing engine. It can produce stripes of different length with one or more repetitions of a pattern. It is a material embodiment, or implementation, of a very simple algorithm that contains a constant data structure and a loop to repeat that structure. There are two input parameters: a starting point (which side of the seal is pressed into the clay first) and a number of rotations (or angle). The result is a stripe of pictures and/or signs. The cylinder seal does not yet have the expressive power of a full programming language. Constancy and repetition are just two elements of such languages, and to be able to program any type of computable function, you need a lot more. But it may be viewed as the first step in this direction.

Some modern printers, e.g., laser printers, still contain a rotating cylinder, but the image printed by the cylinder can be changed on the fly. While the engraver making a cylinder seal was using a medium that he could write to only once, the

---

[22] See Wikimedia Commons, "File:Flickr - Nic's events - British Museum with Cory and Mary, 6 Sep 2007 - 194.jpg," (6 September 2007), available online at URL = <https://commons.wikimedia.org/wiki/File:Flickr_-_Nic%27s_events_-_British_Museum_with_Cory_and_Mary,_6_Sep_2007_-_194.jpg>, for an example of a cylinder seal and the image printed by it (showing a mythological scene involving the god Enki, recognizable from the streams flowing out of his shoulders), with some partial repetition on the edges.

cylinder of a laser printer can be "engraved" with different images again and again. We can use a program written in a programming language to print an image. We might still use a constant content, like an image file. Such a file can be viewed as a special purpose program that can produce just that image on the print surface (the clay of Sumerian cuneiform script and cylinder seals has now been replaced by paper and toner).

But something has remained the same: just as the cylinder seal can be viewed as a data-compressed form of the printed stripes it is used to generate, the program producing printed output can be viewed as a data-compressed form of the printed output. There might be some input parameters influencing what the program does. If you take an output that is longer (in terms of bits or bytes) than the generating program plus its input (again in terms of bits or bytes) than necessarily that output must contain some regularity or pattern because it could be generated by the shorter program. The program (together with its input) may be viewed as a description of that regularity.

We may think of programming languages coupled to printers as some kind of universal cylinder seal, something like a generalization of the seals used by the Sumerians. And just like a stripe unwound from a seal contains some repetition if it is longer than the circumference of the seal, the output of an algorithm contains some regularity (not necessarily a simple repetition, but some kind of order) as long as it is longer than the program generating it. The program can then be viewed as a compressed form of it into which the output can be zipped. The simple rotating seal is a simple special case of this general fact.

We may look at some film of somebody rolling the seal over a clay surface. If we watch that film backwards, we see the picture or character stripe being "parsed" by the seal. The seal now appears as something like a sense organ checking if the stripe is written in the right kind of language or grammar. We could, while rolling the seal over the existing stripe, count the number of revolutions and thus regenerate the compressed version of the information of which the stripe is the decompressed form.

In a similar way, if we replace the special-purpose program of the particular cylinder seal by any algorithm written in a general-purpose programming language, we see that, just like the surface of the seal, the program only contains a limited amount of information. As a result, any output generated by it will show just one particular pattern. If you operate it in reverse mode, it will only be able to parse or accept data fitting into that pattern.

So no single algorithm, with particular input parameters, can be universal, in the sense of being able to produce or accept arbitrary data. Just as every cylinder seal is unique and only produces a limited set of patterns (print stripes of various length with different numbers of complete or partial repetitions), every particular algorithm is also unique and only produces a limited set of patterns, all belonging to a certain type of order.

If we follow theories of the human mind like they are being proposed in the field of "artificial intelligence," aka AI, there would be a particular "intelligent algorithm" implemented in the form of human beings. But such an algorithm could only recognize certain patterns. It would lack universality. Its ability to discover arbitrary patterns existing in reality would be limited.

Likewise, if the method of science could be described in terms of a specific algorithm, science would be limited in its ability to capture the structure of reality to just those patterns that are pre-formed in this algorithm. Science would, in the end, only discover its own structure, not the structure of reality.

If human cognition and human science are universal, on the other hand, they cannot be modelled or described in terms of a single formal theory or algorithm. While knowledge contained in them and produced by them can take the form of algorithms, the description of the human mind must contain mechanisms that enable it to get out of the limits of any particular single formal description. If a part of the mind is describable as an algorithm, this algorithm-changing part of the mind cannot be part of that algorithm itself since what can be calculated inside an algorithm and what can be derived in a formal theory cannot change. The cylinder seal cannot carve or change its own engraving. It looks like the material system we call the human brain has no problem performing such extensions but they cannot be captured in pure algorithmic models.

To summarize this section: each formal theory or algorithm contains a finite amount of information. It can be viewed as a description of a certain pattern or type of regularity in data and as a compressed form of that data. If a stream of data produced or parsed of it is longer than the text of the algorithm it must contain some regularity or order.

We can look at it in terms of information theory: If a stream of data contains order (or redundancy, in terms of information theory), it cannot fill up the information channel completely. It is possible that additional information is added into that information channel and this information could not be parsed by the given algorithm. Extending the algorithm to be able to parse that additional information leads to another algorithm. Any signal stream that can be parsed by that extended algorithm and is longer than the algorithm must again contain some regularity since the algorithm is a compressed form of it and compression requires redundancy (regularity). So again, the information capacity of the information channel cannot be filled completely and there is room for more information (i.e., for the underlying reality that appears as the "sender" of the information to exhibit some other behavior not covered by the given algorithm).

The length of the shortest algorithm required for the description of some signal or data, i.e., the length of the most compressed form of it, is known as the data's *Kolmogorov complexity*. It has been shown that the Kolmogorov complexity of arbitrary data is not Turing-computable.[23] This means that an optimal compression algorithm for arbitrary data does not exist. If it existed, one could just apply it and measure the length of the resulting compressed form of the data. Since information compression means using regularity in data, the fact that an optimal compression algorithm does not exist means that there cannot exist an algorithm that can discover all instances of order in arbitrary data. This means that universal cognition and universal science cannot be described by algorithms, and that algorithmic AI must be limited.

4.3 Computable Total Functions are not Turing-Enumerable

Think of an artificial intelligence system, e.g., a self-driving car. This system has some "senses" (like cameras, radar, microphones, etc.) and some effectors (a motor, brakes, wheels that can change direction, etc.). At each time, it must produce some reaction. The calculation of that reaction can be viewed as the calculation of a function. Input data is mapped onto output data. The system must not get into a state where its data processing goes into a dead loop, so that it would "hang." It must produce a reaction. Generally, such a system might be viewed as computing a "total function," i.e., a function that produces an output for every input (it "halts," i.e., comes to an end state after a limited time, for each input). The computation must always yield some result (perhaps an error message indicating that a solution is not known, but that is also an output).

This view can be extended to learning systems. We can see the total history of inputs as the input. The systems output is then influenced by earlier states.

The "knowledge" of the system can then be viewed as a program for calculating a total function, or as a set of such programs. In each situation, the system can select or construct such a function, based on the input data (and perhaps the history of earlier inputs) and then apply that function to the data.

In order to make thinking about such functions (and proving their properties) easier, we can apply a trick invented by Kurt Gödel (and therefore called

[23] See Wikipedia, "Kolmogorov Complexity," (11 January 2014), available online at URL = <http://en.wikipedia.org/w/index.php?title=Kolmogorov_complexity&oldid=590272459>.

"Gödelization"): it is possible to map any kind of data on natural numbers (the "Gödel numbers" of these pieces of data). Therefore we can, without sacrificing any generality of our argument, restrict ourselves to looking at total functions of natural numbers instead of looking at functions mapping arbitrary types of data onto arbitrary types of data.[24]

So following the AI paradigm, we may think of cognitive processes as processes of information processing, i.e., processes of calculation (this notion also covers the "neuronal networks" currently in fashion in AI: they are algorithms).

In AI-based cognitive science, we can use this also as a model of the mind. In such a model of cognition, we would think of the brain as something comparable to a computer. The justification of such an approach might be disputed, but for the time being, let's assume it is valid. Data are entering the mind through the senses or are the result of previous computations or recalled from memory. The data are transformed by applying some knowledge to it. We can think of the knowledge as consisting of small programs that are applied to data. So we may conceptually divide the process of thinking into two steps (although they might be interwoven into a single unit in reality) that are occurring repeatedly: we select or construct a program, depending on the data or some part of it, and then we apply the program to the data or some part of it.

We can think of the first step as the result of a program that enumerates programs. We enter some data and out pops a program. In the mathematical model, the data are represented by its Gödel number, so the input consists of natural numbers. The programs that are generated are also taking natural numbers (Gödel numbers of input data) as their inputs and produce natural numbers (Gödel numbers of output data) as their output. We can imagine this as a table. Each column corresponds to a function (a piece of knowledge) or the program calculating this function. We may think of the program that produces these functions (or the programs calculating them) as a knowledge base or as a learning program producing new programs, or a combination of these.

As explained above, we assume that the functions are total, i.e., no matter what data we put into such a function, some data will come out after some time (even if that is just an error message or a copy of the input or the like). Our knowledge base does not contain functions that would run forever for some inputs, so that the computer would "hang."

---

[24] See Wikipedia, "Gödel Numbering," (15 July 2017), available online at URL = <https://en.wikipedia.org/w/index.php?title=G%C3%B6del_numbering&oldid=790709724>.

Think of a spreadsheet with infinitely many rows and columns. Each column represents something we can do with data, i.e., a piece of knowledge we can apply to the data in some way. You may think of the numbers in each column as following some pattern described by the program calculating that column (i.e., the program is a compressed form of the data (numbers) in that column).

I am now going to prove that the set of programs calculating computable total functions (on natural numbers) is not enumerable. What this means is that every program we could write that produces such programs (i.e., every knowledge base or learning algorithm) is incomplete.

In order to prove this, we assume the opposite and derive a contradiction. The opposite (negation) of the statement we want to prove is that the set of all programs calculating total functions on natural numbers is enumerable. So we assume that a program is possible that takes the natural numbers as its input and produces as its output all programs calculating total functions. So we could have something like a universal knowledge base. The input would be any data (represented by Gödel numbers). The program would be able to produce any possible program (piece of knowledge) for some input.

Let us call this program K (for "knowledge). For inputs 1, 2, 3, etc. it produces programs k_1, k_2, k_3 etc. calculating total functions. For every program producing a total functions, there is some natural number n so that k_n is that program (since K is complete).

K	k1	k2	k3	k4	k5	k6	k7	k8
1	1	4	2	6	8	3	2	5
2	1	5	4	9	13	6	2	5
3	1	6	6	12	18	9	2	5
4	1	7	8	15	23	12	2	5
5	1	8	10	18	28	15	2	5
6	1	9	12	21	33	18	2	5
7	1	10	14	24	38	21	2	5
8	1	11	16	27	43	24	2	5

You may imagine this as a table, something like an infinitely large spreadsheet. Each column in the spreadsheet corresponds to one function k_x (see the diagram directly above).The fields in each row of that column contain the values of the function. So in column x, row y, you will find the value that program k_x calculates for input y.

Now we use K as the basis to construct a new program calculating a total function. This program works the following way:

- For an input number n, the program k_n is produced by applying K to n. This program k_n is applied to n. To the result, we add 1.

If K is a program, one can write a program that acts like this, so this rule describes a program calculating a computable function. K represents a Turing-enumeration, so it is going to yield a result (a program k_n) for every n after a limited time. k_n is a program calculating a computable total function, so it will yield a value k_n (n) after a limited time. Adding 1 is also a computable operation. So the newly constructed program calculates a computable total function.

You may have noted that this program will take the diagonal of our table. For 1, it gets the value in column 1, row 1. For 2, the value in column 2, row 2 etc.

It then adds 1 to whatever it finds in that square.

Now we have assumed that K is complete. Since the newly defined function is total, there must be some column in our table that contains exactly that function already. Let us say this column has the number m.

The diagonal function goes through every column, so it also goes through column m. So what is contained in the field at column m, row m? What is contained there is the value of $k_m(m)$, i.e., the value of applying k_m to m. And according to the definition of the function, it contains that value, plus 1. So we get $k_m(m) = k_m(m) + 1$. That, however, is impossible. There is no natural number that is equal to itself + 1 (If we subtract $k_m(m)$ on both sides of this equation, we get 0 = 1, which is clearly false). So our assumption that K is complete is leading to a contradiction. This means that no program can exist that can enumerate all programs calculating total functions.

Note that this proof works because we are looking at total functions, so there must be some number in the m-m-square of the table.

Actually this proof, employing the proof method known as (Cantor's) "diagonal method" not only proves that every program enumerating programs calculating total functions is incomplete, it also provides us with a method for constructing a new program calculating a total function that could not be produced by K (such a function that calculates a new element of a set that is not Turing-enumerable, from a subset that is Turing-enumerable, is called a productive function).

Note that instead of adding 1, we could have used any operation that guarantees that the result is not equal to the input. We know that for all numbers x, the result of calculating x+1 is not equal to x. We could have used any other function that, applied to any number, yields an output number different from the input number. For example, we could also add 2 instead of 1, and the proof would work just as well.

Of course, we can add this new program to our enumeration, e.g., by shifting all the columns by one place and adding it as a new column in front (or inserting it in any other position inside the table – we could write some program that chooses a position). This way, we can replace K by a new program K' that produces one function more than K.

Of course, K' is incomplete again: we can apply the same "take the diagonal and modify (e.g., add 1) the result"-method again (and again and again).

We can try to build this diagonalization trick into the program K, so that, for example, every second column will be generated this way. This is possible, but whatever we try, we will not make it to get a complete enumeration, no matter how sophisticated our scheme will become: we can always apply the same diagonal + modify operation "from the outside" and get another program and function not previously covered.

4.4 Limits of Learning-Algorithms

A learning algorithm is a program that, following some fixed set of rules, produces new programs from sets or sequences of input data. Recent progress on the development of some learning algorithms (especially the "deep learning" approach based on "deep," i.e., multi-layered systems of artificial neurons) have led some researchers to the claim that, after a long "AI winter" of lack of success, true AI (artificial intelligence) deserving that name is finally arriving. But are such claims justified? Or is this hype? The proof presented above may be extended to learning programs since we can use entire input histories as inputs. But let me expand on the topic of learning a little bit more.

Initial attempts to build artificial intelligent systems were based on algorithms programmed by people. While such systems could do some interesting special things, like playing checkers and even doing so better than most people, they did not display any ability that could even remotely be called "intelligence." Just as our cars can move faster than we are able to walk, just as our drilling machines are better at making holes than we are with our fingers alone (etc., etc.), such special purpose programs were better than humans at doing some special purpose information processing tasks. But there is nothing special about this; it is one of the reasons why we are using software instead of doing everything by ourselves. It has nothing to do with intelligence. An electronic calculator is better at multiplying numbers than most people, but it is in no way intelligent.

Now the idea that intelligence could instead be achieved by building machines that learn seems plausible at first. The task of producing the special purpose software is shifted to the machine itself. The machine learns from examples and produces the special purpose software by itself. What human programmers provide is not any of the special purpose software but instead a software for producing the special software.

But there is a problem: the learning software is an algorithm itself. As such, it is limited. It can be represented by a finite text, a finite sequence of signs or characters, so it contains only a limited amount of information. The following line of thought (in fact a sketch of a mathematical proof) will show that any learning algorithm is going to be incomplete, i.e., it will not be able to find all the regularity existing in the data presented to it. It will simply not be capable of spotting some patterns in some of the data, i.e., it is going to have systematic blind spots.

Moreover, the amount of programming knowledge contained in such an algorithm, i.e., the number of ways it is able to put together the building blocks of the underlying programming language (basic commands and programming language constructs, artificial neurons and the like) to form new programs, is going to be limited as well. For any given learning algorithm, there are possible programs that another learning algorithm might be able to construct that are out of reach for this particular algorithm.

To see why there always must be such a limit, let us now look back at the proof, sketched in the previous section, that computable total functions are not enumerable by any algorithm. What we have shown there is the following: if we have a program K_1 that produces programs calculating total computable functions, we can always produce another program K_2 that produces all of these programs, plus another one not produced by the previous program.

The proof tells us that each such learning program/knowledge base must be incomplete. We can always construct some knowledge it is not able to produce.

Since the new knowledge can actually be **constructed** from the old formal system (by means of a productive function), there is a process that brings us from one formal description of a cognitive system (think of the "brain" of a robot, for example) to another one. We can always extend the formal system. Could we build this mechanism into the program? Well, this is of course possible and will yield another program K' that will produce additional programs k_x that the original program K could not produce. But this program, no matter how we build it, would also be incomplete. We can apply the same operation to it as before and generate a program it could not produce. If we apply the productive function from the outside, we can apply it to the formal system as a whole. If we build it into the system, the productive system will no longer see the formal system in its totality, i.e., the formal system cannot contain a complete reference to itself (see Ammon 2016).

So, if general intelligence means the ability to produce arbitrary knowledge, a learning algorithm, no matter how impressive its results might be, cannot be generally intelligent. Every algorithm is limited and has some blind spot. Applied to human beings, this means that either we can be described as algorithms, but then we must be limited in our ability to achieve knowledge of the world and there would be things we could not learn in principle because of the way our brain works. Or, we are universal in the sense that we can learn anything. But then we cannot be described completely by algorithms or formal theories.

But can we not solve the problem on the meta-level? Let us assume that we could produce learning programs automatically by some meta-algorithm. If each of them is limited, we might be able to come up with an algorithm instead that produces all such programs automatically. This would be a system that can learn how to learn and produce different ways of learning. Could such a meta-learning algorithm that produces other algorithms automatically be a possible way to achieve general intelligence? Could such an algorithm produce every possible learning algorithm, so that it would not be limited in the way special purpose learning procedures are?

The following proof sketch will show that this is also impossible:

We start again with the assumption that such a program could exist and derive a contradiction from that assumption. Let's assume that we can enumerate all the programs that produce enumerations of total computable functions. So we assume we have a program **K** that produces all possible programs K_1, K_2, K_3, which in turn produce programs k_1, k_2, k_3 etc. calculating total functions, i.e. they enumerate programs computing total functions. Our assumption means that every such enumerating program K_n total that is possible must be in this enumeration produced by **K**.

To make this understandable, we may think of **K** as producing a table. Each column in this table stands for a program K_n that enumerates programs calculating total functions (the k_n in the fields of the table).

K	K1	K2	K3	K4	...
1	k1	k2	k4	k7	
2	k3	k5	k8	...	
3	k6	k9	...		
4	k10	...			
...	...				

Note that here the fields of the table stand for programs calculating total functions, not for the individual values these programs assign to inputs, as in the other proof.

Now, every program k_m that computes any total function must be produced by at least one of the K_n because for each k_m one could easily write a program producing it for one of its inputs (or all of its inputs). Therefore, every program computing a total function must occur somewhere in the table, at least once.

Now, it is possible to bring all the fields of the table into a single sequence. There are several ways of doing this. One is to start with the left upper corner of the table (k_1, green field), then the two fields marked in blue (k2, k3), then the fields marked in yellow, then the orange ones and so on. You can walk through the whole table on such diagonal lines. You can write a computer program that does so (e.g. by means of two loops nested into each other). So from the program **K** you can generate another program K' that enumerates all the programs k_1, k_2, k_3 etc. We can make sure that every program occurs only once in the enumeration if we want to, by checking if it has already been generated before. Since every program calculating a total function must occur somewhere on the table, the resulting enumeration must be complete. However, in the previous proof above, it has already been shown that a complete enumeration of these programs is impossible. Hence, **K** cannot be complete, i.e. programs that produce computable total functions are in turn not enumerable.

Note that we can construct a sequence k_1, k_2, k_3 etc. and apply the "diagonalization + modification"-operation of our first proof to that. This yields a new k'. We can then construct a program producing k' and modify **K** so it would also produce that program. So on this meta-level again, we can go from one algorithm to another, more powerful one, i.e. we can produce a productive function that extends every single given formalism but despite that cannot be used to achieve completeness.

We could construct similar argumentations for meta-meta-meta-programs etc. The argumentations would become more convoluted, but the result will remain the same. It is not possible to construct an algorithm that can produce every possible item of knowledge, or that can produce all possible meta-programs that can generate knowledge, and so on.

If we define a generally intelligent entity (to avoid the term "system here") as an entity that is universal in the sense that it can spot every pattern existing in arbitrary data and can construct every possible program that can embody such patterns, i.e., that it can construct every possible bit of knowledge, such an entity cannot be an algorithm. General learning, which we might understand to be the historical or biographical development of a learning entity in which things not derivable at one stage become derivable at a later one, cannot be an algorithm. It cannot be described completely by a formal theory or algorithm. Such an entity may contain formal theories or algorithms, but it would not be such a formal system itself.

Algorithms or formal theories cannot model or describe their own evolution into other formal theories or algorithms. Everything that is derivable inside a given formal system is so from the outset. Formal systems are ahistorical. The development of knowledge and ideas, and hence human culture, on the other hand, is a historical process in which knowledge is changed. It may be mathematically modelled as a process of applying productive functions to existing knowledge, but this process is not under the complete control of the existing knowledge itself (comparable to mutations in biology which are not under the control of the genes). As a result, the resulting process of evolution of knowledge or culture cannot be described in terms of a single formal theory.

4.5 Shifting the Vantage Point

One Step Beyond

—Cecil Campbell

To understand the issues presented in the previous section, it might be helpful to return once more to the landscape metaphor introduced earlier. Looking at a landscape, you don't see the point from where you are looking, the point where you are standing. That point is part of the landscape, but to see it, together with the landscape, you have to move somewhere else, to a higher peak perhaps, or to a helicopter. If you shift your vantage point, you may be able to see the place where you were standing before, together with the landscape visible from there, but again, you don't see your new point of view.

Something comparable is happening in the "landscape" of formal theories. To see this, we have to get a little bit more technical.

Each formal theory or formal system (analytical space) can be described by a finite-length text containing all the axioms and rules of inference that make it up. There are variations in how formal systems are constructed. Some contain "axiom schemata," some might be thought of as algorithms or programs and all kinds of different formalisms may be used. But in every case, they are describable as finite texts (finite sequences or strings of characters or symbols). The amount of information in a finite text is finite. From such a finite set of initial information, a large and in many cases unlimited set of data or terms or statements can be produced. However, the formal theory you start with contains only a limited amount of information and as a result, everything derivable in it is fixed right from the beginning. From within the formal system, by applying its rules, it is not possible to change it in such a way that it can evolve. It is not possible that within a formal system, something *becomes* derivable that was not derivable before.

If we look at the formal system from the outside, we can look at all its axioms, rules of inference and programs contained in it. We can then apply operations from the outside, change or add something and extend or transform the formal system.

It is possible to build all the operations we apply to make those changes into the formal system. We then get a new, extended formal system that is more "powerful" than the original one. However, this extended formal system again is limited. Everything derivable or computable inside it is derivable or computable right from the beginning; it cannot change inside the new formal system. In this sense, the formal system is static; there is no time in it. It cannot evolve on its own. It can only be evolved from an outside vantage point. Building that outside vantage point into the system leads to a new system that in turn is unable to develop. The vantage point from which that new system can be changed shifts somewhere else.

We can say that a sufficiently rich formal system cannot contain a reference to itself as a whole (see Ammon 2016), in principle. For all such sufficiently rich systems,a reference to a formal system as a whole is always external to the system. Build it into the system and you only get a new formal system for which the point of total reference is again outside it.

The proofs sketched in section 4.4 illustrate this general property of sufficiently rich formal systems. Any program that enumerates such programs, i.e., that produces such programs as its output, is necessarily incomplete, in the sense that it cannot produce all computable total functions. There is an operation that can be applied to it from the outside, from a meta-level, that produces another computable total function not contained in the existing enumeration we started with. This operation can be used to construct an extended, more powerful enumeration program that is an extension of the original one. But that extended enumeration program will in turn be incomplete, and so on.

The operation we apply to do the extension can itself be programmed as an algorithm. It is computable. We can build it into the original program and this would result in a new and more powerful enumeration program that produces all the programs produced by the original enumeration program, and many more, but the proof is valid for this new program again. We can apply the extension operation (what is called a "productive function" in mathematics) to it again from the outside. So building the extension operation into the program has not yielded a complete program. Instead, the outside vantage point again shifted somewhere else. The program cannot contain a reference to itself in its totality so that the gap is closed. Building the reference into it changes it into just another formal system and "squeezes" the external reference somewhere else.

In (Ammon 2016, p. 11 – 16), Ammon presents a proof that is showing that in Gödel's incompleteness proof, the fact that the reference to a formal theory cannot exist inside the formal theory itself is exactly what creates the incompleteness. Roughly speaking, for each formal theory in Gödel's proof a formula can be constructed that is not decidable inside that theory, i.e., it cannot be derived in the theory and its negation can also not be derived. What Ammon is showing is that if the theory is extended by a symbol that represents a reference to the theory as a whole, we get a new, extended theory in which the undecidable formula is provable (the "observing theory" to the original one). However, this new theory in turn has an undecidable formula. A mathematician can always create such an external reference, indicating that a human being cannot be modeled completely in terms of formal systems or reasoning processes.

To come back to the landscape and vantage point metaphor, it is as if from within the original formal theory, we are unable to see the proof of the undecidable formula. But if we shift our vantage point to somewhere else, from where we can see our original vantage point, that proof becomes visible. But there is a new blind spot again.

4.6 Incompleteness and Storage Capacity

There seems to be a connection between the information storage capacity of physical entities and the reach of formal theories describing them.

In (Cubitt et al. 2018), the authors describe the possibility that a material with no band gap could be turned into one with a band gap by adding a single atom and that this switch would be unpredictable. We see here new properties arising when the system grows larger.

Formal theories and algorithms are finite texts. They contain only a limited amount of information (a limited number of bits or bytes) each. We have seen above that if the amount of data described by a formal theory is larger than the size of the formal theory itself, that data must contain some regularity or pattern. If a physical entity becomes larger, either in terms of geometric size or energy content, its information storage capacity goes up. This opens up the possibility of it containing new, additional patterns not described by the original description. As a result, when the information storage capacity is increased, the entity might exhibit properties that are qualitatively new in the sense that they cannot be derived within the theory previously used to describe the entity before it was enlarged.

In the history of life, there is a trend towards increasing size of genomes and towards increased complexity.

In the realm of civilization, we see a trend towards an increase of the memory capacity of cultures, first by the development of language, then by division of work among experts and by mnemonic methods, then by the invention of writing and printing, finally by the introduction of increasingly advanced electronic storage media. This trend towards increased storage capacity seems to be what actually enables cultures to develop new properties.

In all these areas, be it inanimate physical entities, organisms, human societies, or technological entities, an increase in information capacity seems to lead to the possibility of phenomena that are new with respect to a description of an earlier, “smaller” stage.

4.7 How Intelligent Can an Artificial Intelligence Become?

Current AI is algorithm-based, so the systems it produces are not really intelligent, because they lack creativity. I now want to conclude with an argument showing that there is a limit to intelligence. We are not going to see an artificial super intelligence even if we learn how to produce an artificial cognitive proteon. The “singularity” envisaged by some AI and transhumanist folks and the artificial superintelligences they believe are going to emerge in the not so distant future are not going to materialize. What is the reason for this?

Any system that is programmed by an outside programmer and then has a fixed and unchanging program is, in this sense, not intelligent. It contains and applies knowledge but it does not generate new knowledge. So to be intelligent in this sense, an artificial entity would have to be self-programming.

The impressive speed of current computers results from the use of algorithms. Algorithms make use of patterns in the data. That the data is patterned enables the algorithm to use short cuts. It does not need to compare every bit of information with every other bit of information. An algorithm in fact can be viewed as a restriction on which piece of data is combined with which other piece of data. Because algorithms can constrain the combination of data, large amounts of data can be processed in a short time.

In order to produce new knowledge, however, i.e., to discover new patterns in the data, such algorithms cannot be used. An algorithm that constrains which data is compared with or combined with which other data is going to hide unknown patterns behind these restrictions, it has blind spots. So to discover new patterns, one must break out of the restrictions of the existing algorithms. As a result, one has to go back to a stage of more primitive knowledge with a low efficiency.

In such a situation where we do not know an efficient process or algorithm yet, i.e., we do not possess appropriate knowledge yet, we have to try applying different existing methods (algorithms, rules of inference, bits of knowledge) to some data, and see if the result is useful. Each step we make in such a process results in some additional information that we can combine with other information again. If we do not restrict the amount of information we are looking at in such a process at any given time, it will grow very quickly. We may apply some strategy, i.e., some existing algorithm or knowledge, but then we risk not finding some new possibilities. But if we do not do that, the number of possible moves will grow very quickly. We will get a *combinatorial explosion*. The number of possible ways is growing so quickly that no matter what hardware we are using, our computing power is going to be overwhelmed.

In order to avoid combinatorial explosions, we have to restrict the amount of information we are using at any given time to a small number of items. And there is no way to avoid this. A creative process that has a chance to find new and novel knowledge can only be a slow process that processes only small amounts of information at any given time. In order to process large amounts of information, you need an algorithm and that algorithm is always constraining what you do with the information and potentially hiding some structures, creating blind spots. Essentially, an algorithm is a system of constraints on what the computational system will do. As a result, each algorithm has only a limited reach. Some algorithms might enable a system to process large amounts of information, but only by restricting the processing to a small subset of what is possible.

So, creative processes are slow by nature. They cannot be sped up deliberately by adding more processing power. It might be possible to speed up creative processes by using a faster processor, but there is a limit on how fast computers can get, and the processors of our computes have already ceased to become much faster some time ago. You might be able to do some of the "thinking" in parallel, by splitting it up among several processors, but if you let them communicate too much, you are practically creating a combinatorial explosion again.

Moreover, there is no way of guaranteeing the correctness of the results. Creative processes are fallible by nature. If we had a way of knowing in advance what is correct, we could derive new knowledge of any kind, but new knowledge is underiveable from existing knowledge by definition. If you restrict processes of information processing to operations that are guaranteed to be correct, you restrict their ability to find new things. And if you create a system consisting of a "team" of interconnected processes, they might misunderstand each other, and there is no way of preventing that, except by making them part of a fixed algorithm, i.e., by creating a fixed protocol of communication that restricts what these processes can do, and that would again destroy the creativity.

It is not clear how intelligent a cognitive entity can become and where humans or teams of humans are with respect to the possible maximum intelligence. However, our brains are able to process large amounts of information simultaneously, so the processing power of our networks of neurons is not the limiting factor here. The limit of "7 items at a time, plus or minus two" that seems to exist according to cognitive psychology is not a limit of the neuronal "technology" our brains are using; instead it seems to be a limit that is there as a matter of principle. Put more pieces into the game at any given time, and the number of ways you can combine them explodes exponentially. So probably artificial intelligent entities, should they ever be created, might be thinking slightly faster than we do, but they would not really be much more intelligent. Computers are going to surpass our abilities on any task that can be formalized (that is what they have always been doing, even in the case of simple calculations), but we are not going to see any artificial super intelligence. There is no singularity ahead.[25] This, by the way, provides another argument against the computational metaphysics presented in (Bostrom 2003): the super intelligences envisioned there are not possible.

5. Concluding Summary

To sum it all up, let me try to extract the main ideas of this essay and elaborate some slightly further, skipping over the many loose ends at the same time.

[25] See Wikipedia, "Technological Singularity," (25 November 2018), available online at URL = <https://en.wikipedia.org/w/index.php?title=Technological_singularity&oldid=870578683>.

To describe an entity exactly and completely means that it is possible to describe it in terms of a formal theory or algorithm (the notions of "formal theories" and "algorithms" are equivalent). Mathematical objects are known for which such a complete and exact description is not possible (e.g., the set of total computable functions of natural numbers discussed above). Such entities, for which a complete formal description cannot be achieved, are called "proteons" or protean entities. In contrast to proteons, objects for which a complete exact description is possible are called "systems" or systematic entities. So in the context of the terminology introduced here, the term "system" is restricted to what is formalizable.

Computability theory describes the boundary between what can or cannot be described in terms of algorithms. It therefore also describes the boundary of what can or cannot be described by formal theories. We may therefore also call it "formalizability theory." Complexity theory describes which algorithm can be practically implemented and executed and which ones cannot because the resources needed (time, storage space) would exceed anything that can be made practically available. So together, computability theory and complexity theory describe the entities that can be described exactly in practice. For practical reasons, it does not make a difference whether something is non-computable in principle or whether it is practically non-computable for reasons of complexity theory. In both cases, we can use the term "proteon" for such an entity (and one could distinguish here between "theoretical" and "practical" proteons).

Physicists have found examples of physical entities that are theoretical proteons. Such entities have more properties than can be derived or computed in any single formal theory about them, i.e., there are more true statements about such entities than can be derived in any single formal theory about them. If a formal theory of such an object is given, then there are physical processes that can move the object out of the scope of that theory. For example, a physical process like adding an atom to a solid object can change some electronic properties of that object in unpredictable ways. It is possible to extend a particular theory about such objects, so the resulting expanded theory covers more cases, but this extended theory would be incomplete again.

The existence of such cases in physics means that physics as a whole in not computable. Reality as a whole cannot be described in terms of any single formal theory. So even if we could find a general theory of everything, it would have to be computationally incomplete, so that new mathematical knowledge would have to be added again and again, extending the formal theories used to describe reality. As a result, scientists will always have to deal with a multitude of special theories for special sub-sets of reality.

The existence of such cases in physics also means that computability is not the same as possibility. Reality is not restricted to computable processes. Physical processes happen according to natural laws but they are not processes of computation and are thus not limited to the computable. Moreover, physical processes that cannot be calculated effectively for reasons of complexity (i.e., where the resources to do such a calculation would exceed the resources that can be made available by any possible technology) can happen, since physical processes are not calculations (protein folding seems to be an example of this).

The restrictions of computability theory are limited to formal systems, not to physical systems. This means that physical entities may be possible that perform processes of information processing that exceed the limits of algorithms that can be implemented on Turing machines. I call such an entity a cognitive proteon (in Ammon's terminology, this is a "creative system"). A basic thesis of this essay is that human beings can be viewed as cognitive proteons. The question whether all mental processes can be covered by the notion of "information processing" is an open one, but to the extent that they can, the hypothesis put forward here (taken over from Ammon) is that the capabilities of human beings exceed those of algorithms. A complete formal/algorithmic theory of human cognition is impossible. In this respect, my view differs from AI and AI-based cognitive science, whose models (including "neuronal network" and "deep learning" based approaches) can be described as algorithms. Human cognition, by contrast, is protean. While parts of our knowledge, at any time, can be described in terms of algorithms, the totality of cognition cannot. It contains processes, called "creative processes" by Ammon, by which the systematic or algorithmic structures can be changed. These processes cannot be captured completely by any algorithm or formal theory. Each single one could be described algorithmically in hindsight but no algorithm can be given that describes them in their totality.

Systematic sub-sections of knowledge, together with the objects they refer to, are called "analytical spaces" (a concept also taken over from Ammon). The partial scientific theories mentioned above are analytical spaces, but the concept can be extended to everyday knowledge. Inside cognition, a framework of basic methods of thinking, representations (not necessarily conceptual), and categories, or basic cognitive structures, might develop and such a framework might be quite stable and fundamental for many cognitive processes, but it is the result of historic processes, both genetic and cultural, and it can be modified, changed and extended. The processing of information can be influenced by other information. The influencing information then plays the role of programs, and it can be thought of as active. In this way, the structure of the cognitive processes can be changed in any way, in a way that resembles programming.

Such cognitive structures can be viewed as a kind of analytical spaces. Since the creative processes creating cognitive structures and knowledge in general are not processes of formal derivation or calculation, creative processes are open to error. There is no guarantee that they will yield correct results. So the totality of analytic spaces contains structural breaks, i.e., there are inevitably leaps and uncertain hypotheses. A complete unification of all analytical spaces is impossible, i.e., it is impossible to establish a single framework that guarantees certainty. There is no synthetic a priori knowledge. The generation of knowledge has a quasi-rationalistic basis at any time, since some framework of pre-existing structures is always present[26], but this basis is itself the result of historical processes and has to prove its value empirically. Thus cognition advances empirically from simple, less comprehensive, and less efficient sets of analytical spaces to more advanced sets of analytical spaces, through a series of divisions, unifications, and modifications of analytical spaces and the generation of new ones. The protean reality produces surprises, or anomalies, i.e., new information that could not be predicted by the set of analytical spaces present at a given time. By the assimilation or integration of this new information, new analytical spaces form or existing ones are modified.

If an algorithm can describe a set of data completely (i.e., generate or parse it without loss of information), and if the data set is larger (in terms of storage space) than the algorithm, then the algorithm can be viewed as a compressed form of the data set. Being compressible requires having regularity, and the algorithm can then be viewed as a description of that regularity. For each kind of regularity described by a particular algorithm, different patterns can be constructed that do not show that particular kind of regularity. So no single algorithm can describe all kinds of patterns or structures, and each single algorithm can only produce or parse a limited type of structures or patterns. Algorithms and the data described by them form analytical spaces. A cognitive entity capable of discovering arbitrary patterns in arbitrary data therefore cannot be an algorithm. It must instead be able to develop arbitrary new algorithms. Each single algorithm (including learning algorithms) has systematic blind spots. As a result of this, if human cognition and science are universal in the sense that they are capable of discovering arbitrary structures of reality, then they cannot be completely systematic (algorithmic) but must be protean, at least if reality is itself a proteon. Otherwise, there would be systematic limits to what we can understand arising from the (fixed) structure of our cognition, i.e., systematic blind spots of our cognitive capabilities. My thesis is that human cognition—and also science as a collective cognitive process—is creative in the sense that such blind spots do not exist or that they can always be removed. So while reality cannot be described completely, a stepwise refinement and increasing approximation is possible.

---

[26] Ammon speaks of a "reflection base," a set of simple cognitive structures that form the basis of creative processes; see Ammon 1987.

There is a traditional separation between “hard sciences” that attempt formalizations of their fields, and other areas of knowledge, like the humanities, where the use of “the scientific method” is difficult or impossible. As a result, there is also a division into what is generally known (following C.P. Snow) as the “two cultures.” But if physical reality is a proteon, and scientists have to work with partial approximations (as they actually do in practice) the difference between the two cultures and the two separate “academic areas” of “science” and “scholarship” becomes blurred. The division can be bridged, not by reduction of the humanities to hard science, which is impossible, but by scientists embracing the protean nature of reality and the limitations of exact, formal methods. The difference between the two areas is therefore not an essential one. The hard sciences in this view appear as containing very large analytical spaces, in which anomalies can be pushed beyond the horizon, so to speak, so that it is possible to move around inside them in a systematic way. On the other hand, in more complex areas like the study of human culture (and science itself as part of it), people are forced always to “juggle” with several small analytical spaces at the same time, and constantly move to a meta-level of reasoning about these analytical spaces and the shifts between them. In scientific and philosophical praxis, however, both modes of thinking, the systematic and the creative, belong together and scientists and scholars have to switch between the two. In the large analytical spaces of classical science, the need to do so arises less often and is most prominent during a scientific crisis, but as scientists are moving into increasingly complex areas, they are increasingly going to find themselves in situations where the creative mode of thinking is inevitably necessary.

I suggest that this meta-level of thinking about analytical spaces and their limits, and of criticizing, analyzing and also modifying them, is the real realm of theoretical philosophy. In this sense, one could say, in a slightly simplifying manner, that science is about systems and philosophy is about proteons. Since the demarcation line between systems and proteons is described by computability theory and complexity theory, these mathematical theories may then be viewed as core components of theoretical philosophy.

References

(Ammon 2016) Ammon, K., “Informal Physical Reasoning Processes,” (2016), available online at URL = <http://arxiv.org/abs/1608.04672v1>; downloadable .pdf available online at URL = <https://arxiv.org/pdf/1608.04672v1.pdf>.

(Ammon 2013) Ammon, Kurt: “An Effective Procedure for Computing ‘Uncomputable’ Functions,” (2013), available online at URL = <https://arxiv.org/abs/1302.1155>; downloadable .pdf available online at URL = <https://arxiv.org/pdf/1302.1155.pdf>.

(Ammon 1987) Ammon, Kurt: “The Automatic Development of Concepts and Methods,” (Doctoral Dissertation, University of Hamburg, Dept. of Computer Science, 1987).

(Bostrom 2003) Bostrom, N., “Are You Living in a Computer Simulation?,” *Philosophical Quarterly* 53 (2003): 243-255, also available online at URL = <https://www.simulation-argument.com/simulation.html>.

(Cubitt et al. 2018) Cubitt, T., Perez-Garcia, D., and Wolf, M., “The Unsolvable Problem,” *Scientific American* 319 (2018): 20-29.

(Cubitt et al., 2015) Cubitt, T., Perez-Garcia, D., and Wolf, M., “Undecidability of the Spectral Gap", *Nature* 528 (2015): 207 to 211, preprint available online at <https://arxiv.org/abs/1502.04573v3>; downloadable .pdf available online at URL = <https://arxiv.org/pdf/1502.04573v3.pdf>.

(Deutsch 1985) Deutsch, D., “Quantum Theory, the Church-Turing Principle and the Universal Quantum Computer,”(July 1985), available online at URL = <http://www.ceid.upatras.gr/tech_news/papers/quantum_theory.pdf>.

(Dilthey 1983) Dilthey, W. *Gesammelte Schriften* (Göttingen: Vandenhoeck & Ruprecht, 1983), vol. XIX.

(Everett 2005) Everett, D. “Cultural Constraints on Grammar and Cognition in Piraha,” *Current Anthropology* 46 (2005), also available online at URL = <https://www1.icsi.berkeley.edu/~kay/Everett.CA.Piraha.pdf>.

(Fischer 2009) Fischer, J., “Philosophische Anthropologie: Eine Denkrichtung des 20. Jahrhunderts,“ (Freiburg/München, 2009).

Borderless Philosophy 2 (2019): 117-172.

(Fraenkel 1993) Fraenkel, A.S., "Complexity of Protein Folding," *Bulletin of Mathematica Biology* 55 (1993): 1199, available online at URL = <https://link.springer.com/article/10.1007/BF02460704>.

(Hempel and Oppenheim 1948) Hempel, C., and Oppenheim, P., "Studies in the Logic of Explanation," Philosophy of Science 15 (1948): 135-175, also available online at URL = <http://www.sfu.ca/~jillmc/Hempel%20and%20Oppenheim.pdf>.

(Keller 2017) Keller, A., "Computability and Physics,"Against Professional Philosophy (25 October 2017), available online at URL = <https://againstprofphil.org/2017/10/25/computability-and-physics/>.

(Kuhn 1996) Kuhn, T., *The Structure of Scientific Revolutions* (3rd edn., Chicago, IL: Univ. of Chicago Press, 1996).

(Marquard 2004) Marquard, O., "Die Philosophie der Geschichten und die Zukunft des Erzählens," in K.-H. Lembeck (ed.), *Geschichte und Geschichten: Studien zur Geschichtenphänomenologie Wilhelm Schapps* (Würzburg: Königshausen & Neumann, 2004), pp. 45 – 56.

(Snow 1959) Snow, C.P., *The Two Cultures* (Cambridge: Cambridge Univ. Press, 2001), originally published in 1959.

(Weizsäcker 1977) Weizsäcker, C. F. Von, *Der Garten des Menschlichen* (München: Hanser, 1977).

Wikipedia, "Gödel Numbering," (15 July 2017), available online at URL = <https://en.wikipedia.org/w/index.php?title=G%C3%B6del_numbering&oldid=790709724>.

Wikipedia, "Kolmogorov Complexity," (11 January 2014), available online at URL = <http://en.wikipedia.org/w/index.php?title=Kolmogorov_complexity&oldid=590272459>.

Wikipedia, "Proceedings of the Royal Society A," 400 (1818): 97–117, available online at URL = <https://en.wikipedia.org/wiki/Proceedings_of_the_Royal_Society#Proceedings_of_the_Royal_Society_A>.

Wikipedia, "Proteus," (2018), available online at URL = <https://en.wikipedia.org/w/index.php?title=Proteus&oldid=857609946>.

Borderless Philosophy 2 (2019): 117-172.

Wikipedia, "Technological Singularity," (25 November 2018), available online at URL = <https://en.wikipedia.org/w/index.php?title=Technological_singularity&oldid=870578683>.