Zuse's thesis, Gandy's thesis, and Penrose's thesis

Ver. 12

Zuse's Thesis, Gandy's Thesis, and Penrose's Thesis

Jack Copeland, Oron Shagrir, Mark Sprevak

1. Introduction

Computer pioneer Konrad Zuse (1910-1995) built the world's first working program-controlled general-purpose digital computer in Berlin in 1941. After the Second World War he supplied Europe with cheap relay-based computers, and later transistorized computers. Mathematical logician Robin Gandy (1919-1995) proved a number of major results in recursion theory and set theory. He was Alan Turing's only PhD student. Mathematician Roger Penrose (1931-) is famous for his work with Stephen Hawking. What we call Zuse's thesis, Gandy's thesis, and Penrose's thesis are three fundamental theses concerning computation and physics.

Zuse hypothesized that the physical universe is a computer. Gandy offered a profound analysis supporting the thesis that every discrete deterministic physical assembly is computable (assuming that there is an upper bound on the speed of propagation of effects and signals, and a lower bound on the dimensions of an assembly's components). Penrose argued that the physical universe is in part *un*computable. We explore these three theses. Zuse's thesis we believe to be false: the universe might have consisted of nothing but a giant computer, but in fact does not. Gandy viewed his claim as a relatively apriori one, provable on the basis of a set-theoretic argument that makes only very general physical assumptions about decomposability into parts and the nature of causation. We maintain that Gandy's argument does not work, and that Gandy's thesis is best viewed, like Penrose's, as an open empirical hypothesis.

2. Zuse's thesis: the universe is a computer

Zuse's book *Rechnender Raum* ("Space Computes") sketched a new framework for fundamental physics (Zuse 1969). *Zuse's thesis* states that the physical universe is a digital computer—a cellular automaton.

The most famous cellular automaton is the *Game of Life* (GL), invented in 1970 by John Conway (Gardner 1970). GL involves a grid of square cells with four transition rules, such as "If a cell is on and has less than two neighbors on, it will go

off at the next time step", and illustrates an interesting phenomenon: complex patterns on a large scale may emerge from simple computational rules on a small scale. If one were to look only at individual cells during the GL's computation, all one would see is cells switching on and off according to the four rules. Zoom out, however, and something else appears. Large structures, composed of many cells, grow and disintegrate over time. Some of these structures have recognizable characters: they maintain cohesion, move, reproduce, interact with each other. They are governed by their own rules. To discover these higher-order rules, one often needs to experiment, isolating the large structures and observing how they behave under various conditions.

The behavior can be dizzyingly complex. Some patterns, consisting of hundreds of thousands of cells, behave like miniature universal Turing machines. Larger cellular patterns can build these universal Turing machines. Yet larger patterns feed instructions to the universal Turing machines to run GL. These in-game simulations of GL may themselves contain virtual creatures that program their own simulations, which program their own simulations, and so on. The nested levels of complexity that can emerge on a large grid are mind-boggling. Nevertheless, everything in GL is, in a pleasing sense, simple. The behavior of every pattern, large and small, evolves exclusively according to the four fundamental transition rules. Nothing happens in GL that is not determined by these rules.

Zuse's thesis is that our universe is a computer governed by a small number of simple transition rules. Zuse suggested that, with the right transition rules, a cellular automaton would propagate patterns, which he called *Digital-Teilchen* (digital particles), that share properties with real particles. More recently, Gerard 't Hooft, Leonard Susskind, Juan Maldacena, and others have suggested that our universe could be a hologram arising from the transformation of digital information on a two-dimensional surface (Bekenstein 2007). 't Hooft says: "I think Conway's Game of Life is the perfect example of a toy universe. I like to think that the universe we are in is something like this" ('t Hooft 2002).

GL's four transition rules correspond to the fundamental "physics" of the GL universe. These are not the rules of our universe, but perhaps other transition rules are—or perhaps the universe's rules are those of some other type of computer: David Deutsch and Seth Lloyd suggest that the universe is a quantum-mechanical computer instead of a classical cellular automaton (Deutsch 2003, Lloyd 2006). If Zuse's thesis

is right, then all physical phenomena with which we are familiar are large-scale patterns that emerge from the evolution of some computation operating everywhere in the universe at the smallest scales. A description of that computation would be a unifying fundamental physical theory.

Should we believe Zuse's thesis? One can get an idea of how much credence to give it by considering what would need to be true for the thesis to command rational belief. There are three big problems that a defender of Zuse's thesis needs to overcome. The first is the *reduction problem*: show that all existing physical phenomena, those with which we are familiar in physics, could emerge from a single underlying computation. The second is the *evidence problem*: provide experimental evidence that such an underlying computation actually exists. The third is the *implementation problem*: explain what possible hardware could implement the universe's computation.

Our focus is on the implementation problem (we discuss the reduction problem and the evidence problem in Copeland, Sprevak and Shagrir 2017). What is the hardware that implements the universe's computation? A computation requires some hardware in which to occur. As we all know, the computations that a laptop carries out are implemented by electrical activity in silicon chips and metal wires. The computations in the human brain (if such there are) are presumably implemented by electro-chemical activity in neurons, synapses and their substructures. In Conway's original version of GL, the computation is implemented by plastic counters on a Go board. Notably, the implementing hardware, the medium that every computation requires, must exist in its own right. The medium cannot be something that itself emerges from the computation as a high-level pattern. Conway's plastic counters cannot emerge from GL: they are required in order to play GL in the first place. What then is the medium in the case of the universe?

According to Zuse's thesis, all phenomena with which we are familiar in physics emerge from some underlying computation. The medium that implements this computation cannot be something that we already know in physics (for example, the movement of electrons in silicon) since, by Zuse's thesis, that would be an emergent pattern from the underlying computation. The medium must be something outside the realm of current physics. But what could that be? In what follows we present four options. None leave us in a happy place.

The first option is *weird implementers*. This option boldly asserts that something outside the current catalogue of physical entities, and hence 'weird', implements the universe's computation. In principle, a weird implementer could be anything: ectoplasm, angelic gloop, or the mind of God. A weird implementer could also emerge from another computation that has its own weird implementers, which in turn emerge from another computation, and so on. Different versions of the weird implementers response posit different specific entities to implement the universe's computation. Weird implementers are objectionable not because we can already rule them out based on current evidence but because they offend principles of parsimony and the usual scientific standards on evidence. Positing a specific new type of entity should be motivated. If it can be shown that positing some specific type of entity does essential explanatory work for us – work that cannot be done as well any other way – that would be a good argument for its existence. But positing a specific weird implementer merely to solve the implementation problem seems ad hoc and unmotivated.

An alternative version of the weird implementers response is to repurpose some *non-physical* entity, which we already know to exist (so avoiding the charge of adding to our ontology), as hardware for the physical universe. What would remain is to show that this entity does indeed stand in the implementation relation to the universe's computation. Max Tegmark has a proposal along these lines (Tegmark 2014). Tegmark's 'Mathematical Universe Hypothesis' claims that the implementing hardware of the physical universe consists in abstract mathematical objects. The existence of abstract mathematical objects is, of course, controversial. But granted that one accepts (on independent grounds) that those objects exist, Tegmark's idea is that those objects can be repurposed to run the universe's computation. Among the mathematical objects are abstract universal Turing machines. Tegmark proposes that the physical universe is the output of an abstract universal Turing machine run on random input. A similar suggestion is made in Schmidhuber (2013).

Many objections could be raised to this proposal. The most relevant for us is that abstract mathematical entities are not the right kind of entity to implement a computation. Time and change are essential to implementing a computation: computation is a process that unfolds through time, during which the hardware undergoes a series of changes (flip-flops flip, neurons fire and go quiet, plastic

counters appear and disappear on a Go board, and so on). Abstract mathematical objects exist timelessly and unchangingly. What plays the role of time and change for this hardware? How could these Platonic objects change over time to implement distinct computational steps? And how could one step "give rise" to the next if there is no time or change? Even granted abstract mathematical objects exist, they do not seem the right sort of things to implement a computation.

The second solution is *instrumentalism* about the underlying computational theory. This replays Mach's treatment of nineteenth-century atomic theories in physics. Mach argued that atomic theories, while predictively successful, do not aim at truth: the atom 'exists *only* in our understanding, and has for us only the value of a *memoria technica* or formula' (Mach 1911: 49). A scientific theory need not aim at giving a true description of the world. Its value may rather lie in the instrumental goods it delivers: making accurate predictions, unifying diverse results, aiding calculation, grouping phenomena together in perspicuous ways, and prompting useful future enquiries.

If we are instrumentalists about the computational theory that underlies our universe then we avoid the implementation problem. An instrumentalist does not care about the computational theory being true, only about its instrumental utility. An instrumentalist sees no problem in positing things that do not exist (the Coriolis force, mirror charges, positively-charged holes, etc.) to achieve her ends. The implementers of the universe's computation could therefore, for an instrumentalist, be anything real or imagined. The implementers could even be notional: assumed for the nonce to generate predictions. An instrumentalist would lose no sleep over the existence or non-existence of implementers as she has no investment in the theory being true.

Instrumentalism may be a reasonable attitude to adopt towards some scientific theories (for example, geocentric planetary theories still used for navigation but known to be false). However, it takes a strong stomach to be an instrumentalist about a fundamental physical theory. Zuse's thesis is usually couched as a claim about the *true nature* of the universe: the universe is a giant computer. Our question was why we should believe this. The instrumentalist responds by changing topic: not by showing that Zuse's thesis is credible, but by arguing that it is useful (and even that much has not yet been shown).

The third solution is *anti-realism* about the fundamental physical theory. Anti-realism is the idea that some features of the universe that may appear to be objective features are, in fact, mind dependent. Zuse's thesis claims that a computation takes place. This claim is presumably made true by the implementers of the computation behaving one way rather than another—by them satisfying a specific pattern described by that computation. On a Go board with plastic counters, whether GL is taking place or not is made true by the implementers behaving in one way rather than another: if a plastic counter is on a specific square at a given moment, the cell is "on"; if it is not, the cell is "off". But what if there were no implementers and the decision about whether an implementer is behaving *this* way rather than *that* way lay inside the head of an agent? GL does not need to involve a Go board and plastic counters. It could for example take place by the agent keeping track of appropriate sequences of "yes" or "no" decisions that settle the question of whether a specific counter is on a specific square. Like Dr B in Stefan Zweig's *Schachnovelle*, the agent might generate a sequence of decisions that implement GL in her head. This may not be easy or convenient, but there is no reason it could not be done. In this case, the hardware that implements the computation would be *mind dependent*.

There is nothing problematic about this considered as a proposition about GL. The anti-realist tries to play the same trick for the computation postulated by Zuse's thesis. John Wheeler's "It from bit" doctrine can be viewed as a move in this direction:

> [T]hat which we call reality arises in the last analysis from the posing of yes-no questions and the registering of equipment-evoked responses; … all things physical are information-theoretic in origin and … this is a *participatory universe*. (Wheeler 1990: 5)

> We are participators in bringing into being not only the near and here but the far away and long ago. (Wheeler 2006)

The idea is that the fundamental informational "yes"/"no" states that underlie the physical universe are somehow generated by observers. It is not clear how broad the category of "observer" is: whether it includes simple devices like photographic plates as well as conscious humans. But no matter how broad or narrow this class, the anti-realist solution to the implementation problem should produce a sense of disquiet.

As was mentioned above, the hardware that implements a computation cannot emerge from that computation. But this is precisely what is required here. An anti-realist says that the implementation of the universe's computation lies in the registering of a sequence of bits by agents or other observers. But the anti-realist solution also requires that those agents and other observers be physical parts of the universe—they need to be to interact causally with the rest of the universe. Therefore, agents and other observers play a dual role: implementing the universe's computation *and* being among the high-level products that emerge from that computation. This contradicts our principle that the hardware that implements a computation cannot emerge as a high-level product from that computation. We have no model of how implementation could work in this case. Anti-realism about computations that take place *inside* the universe (such as GL) is unproblematic. Anti-realism about the computation that generates the *entire* physical universe (including all agents and other observers) seems mysterious and incoherent. At best, it would require significant reworking of existing ideas of implementation.

The fourth solution to the implementation problem is *epistemic humility* about the implementers. This is the suggestion that we trim our ambitions regarding knowledge of the implementers. We know that *something* must implement the universe's supposed computation, but according to this response we say that we know nothing—and can know nothing—about that shadowy substratum. Our proper aim should be to describe the universe's computation; we should remain silent about the nature of the implementing medium. Unlike the weird implementers option, epistemic humility makes no positive claim about the specific nature of the implementers other than that some implementer must exist. Unlike instrumentalism, epistemic humility says that Zuse's thesis aims at delivering truth and not just instrumental benefits. Unlike anti-realism, epistemic humility makes no claim that minds or observers are part of the implementing medium.

There are precedents for this kind of humility. Henri Poincaré argued that science can tell us only about the "true relations" between "real objects which Nature will hide forever from our eyes" (Poincaré 1902: 161). Bertrand Russell argued that science can tell us only about the structure of matter, not about its "intrinsic character" (Russell 1927: 227). These expressions of epistemic humility share the idea that the world contains some sort of shadowy substratum (although neither author says that

that the substratum implements a computation). Following this line of thought, an advocate of Zuse's thesis might argue that we should not be troubled about committing to the view that a substratum exists—even if knowledge of the nature of that substratum is forever beyond us.

The problem with epistemic humility is that it does not so much answer the implementation problem as admit that we cannot answer it. If one was motivated by the implementation problem at all, one is unlikely to find this a satisfying solution. If the universe is a computer, one might feel that we should be able to say something positive about the implementing medium. Epistemic humility requires that we surrender all ambitions on this score.

Epistemic humility deals with the implementation problem by saying that we can never solve it, instrumentalism changes the topic from truth to usefulness, anti-realism is of dubious coherence, and proponents of weird implementers either shoehorn unsuitable entities into the role of implementers or else indulge in unjustified speculation. These options are not meant to be exhaustive and the considerations raised are not intended to refute Zuse's thesis. But we have at least put some hard questions on the table (and we say more in Copeland, Sprevak and Shagrir 2017).

One potential route forward for advocates of Zuse's thesis is to combine instrumentalism, anti-realism and epistemic humility in a way described by Dennett (1991) and Wallace (2003).[1] On such a view, whether something counts as real or not depends on how useful it is to admit it into our ontology. If a computational theory in fundamental physics were to prove sufficiently useful, then, on this view, we should regard the computation described by the theory as real and adopt an attitude of epistemic humility towards the implementing medium. It remains to be seen, of course, how useful Zuse's thesis will prove in fundamental physics.

Even if the universe is not a computer it may nevertheless be comput*able*. We turn next Gandy's thesis.

3. Gandy's thesis: Turing computability is an upper bound on the computations performed by discrete deterministic mechanical assemblies

[1] Thanks to Michael Cuffaro for this suggestion.

In his 1980 article "Church's Thesis and Principles for Mechanisms", Gandy advanced and defended a proposition that he termed "Thesis M": "What can be calculated by a machine is computable" (1980: 124).

Gandy said that by *computable* he means "computable by a Turing machine", and he takes the objects of computation to be functions over the integers (or other denumerable domains). It is less clear what he meant by *calculation* and *computation* (we ourselves will use these terms interchangeably) and by *machine*. He said that he was using "the fairly nebulous term 'machine'" for the sake of "vividness", and he made it evident that discrete deterministic *mechanical* assemblies are his real target, where the "only physical presuppositions" made about a *mechanical* system are that there is "a lower bound on the linear dimensions of every atomic part" and "an upper bound (the velocity of light) on the speed of propagation of changes" (1980: 126). We will refer to discrete deterministic mechanical assemblies as DDMAs. Gandy emphasized that the arguments in his paper apply only to DDMAs and not to "*essentially* analogue" systems, nor systems "obeying Newtonian mechanics" (1980: 126, 145). His thesis—which we call Gandy's thesis—is that the functions able to be computed by DDMAs are Turing computable.

Like his teacher Turing, Gandy took an axiomatic approach to characterizing computation. But whereas Turing's classic 1936 paper gave an analysis of *human* computation (Turing 1936; see further Copeland 2004, 2017), Gandy's aim was to provide a wider analysis. He pointed out that Turing's analysis does not apply to machines in general: Turing assumes, for instance, that the computer (a human being) "can only write one symbol at a time", an assumption that clearly does not apply to parallel machines, since these can change "an arbitrary number of symbols simultaneously" (1980: 124-5). Gandy formulated the general concept of a DDMA in terms of precise axioms, which he called Principles I – IV. These four axioms define a set of mechanisms—"Gandy machines"—and Gandy proved that the computational power of these mechanisms is limited to Turing computability (a simplified version of the proof is provided by Sieg and Byrnes 1999).

Principle I, which Gandy referred to as giving the "form of description", sets out a format for describing DDMAs. A DDMA is described by an ordered pair $<S,F>$, where S is a potentially infinite set of states and F is a state-transition operation from S_i to S_{i+1} (for each member S_i of S). Gandy chose to define the states in terms of

subclasses of the hereditarily finite sets (HF) over a potentially infinite set of atoms (closed under isomorphic structures). These subclasses are termed "structural classes"; and the state-transition operation is defined in terms of structural operations over such classes. Putting aside the technicalities of Gandy's presentation, Principle I can be approximated as:

> Principle I: Any DDMA M can be described by an expression $\langle S,F \rangle$, where S is a structural class, and F is a transformation from S_i to S_j. Thus, if S_0 is M's initial state, then $F(S_0)$, $F(F(S_0))$,… are its subsequent states.

Each (non-atomic) state S_i of S is assembled from parts, and these can be assemblies of other parts, etc. Principles II and III place boundedness restrictions on the structure of the states. They can be expressed informally as:

> Principle II: For each machine, there is a finite bound on the complexity of the structure of its states. (In Gandy's terminology, this comes down to the requirement that the states of a machine are members of a fixed initial segment of HF.)

In GL, for example, the grid can be arbitrarily large but the complexity of the structure of each state is very simple and can be described as a list of pairs of cells—or, more generally, as a list of lists of cells, since each listed pair of cells is itself a list of cells. In general we can picture a Gandy machine as storing information in a hierarchical way, such as lists of lists (Gandy 1980: 131), but Principle II lays down that for each machine there is always a finite bound on the structure of this hierarchy.

> Principle III: There is a bound on the number of types of basic parts (atoms) from which the states of the machine are uniquely assembled.

For example, the grid of GL can be assembled from pairs of consecutive cells and their symbols (e.g. ('on', 'off'), ('on', 'on'), etc). We need only a limited number of pairs like these to construct any configuration of the grid.

Principle IV puts restrictions on the structural operations that can be involved in state transitions: each state transition must be determined by the *local* environments of the parts of the assembly that change in the transition. Gandy called this the "principle of local causation" and described it as "the most important of our principles" (1980: 135). He explained that the axiom's justification lies in the two "physical presuppositions" governing mechanical assemblies (mentioned above). If the propagation of information is bounded, then in bounded time an atom can transmit

and receive information in a bounded neighborhood; and if there is a lower bound on the size of atoms, then the number of atoms in this neighborhood is bounded. Taking these together, we can informally express the principle as follows:

> Principle IV: The parts from which $F(S_i)$ is assembled are causally affected only by their bounded "causal neighbourhoods": the state of each part is determined solely by its local neighbourhood.

For example, in GL the grid is assembled from parts—cells—each of which is either 'on' or 'off' at any given moment. A cell's state—'on' or 'off'—is determined only by the bounded causal neighbourhood consisting of its eight adjacent cells.

Gandy's proof that any assembly satisfying Principles I – IV is Turing computable goes far beyond the (relatively trivial) textbook reduction of the actions of some number of Turing machines working in parallel to the action of a single Turing machine. There are Gandy machines with arbitrarily many processing parts that work on the same regions (e.g. printing on the same region of tape), and also Gandy machines whose state-transitions involve simultaneous changes in an unbounded number of parts. In GL, for example, there is no upper bound on the number of cells that are simultaneously updated.

To what extent does Gandy's analysis capture machine computation? Wilfried Sieg contends that Gandy provided "a characterization of computations by machines that is as general and convincing as that of computations by human computors given by Turing" (Sieg 2002: 247). We challenge Sieg's contention. It is doubtful that Gandy's analysis even encompasses all cases of *physical computation*, not to mention computation carried out by other, notional, machines. Moreover, even Gandy himself thought that not all physical computing machines lie within the scope of his characterization; and for this reason he explicitly distinguished between "mechanical devices" and "physical devices", saying that he was considering only the former (Gandy 1980: 126). As we explained above, Gandy said that his analysis aims only at machines conforming to the principles of Relativity, and he expressly excluded some machines that obey Newtonian mechanics—e.g. machines involving "rigid rods of arbitrary lengths and messengers travelling with arbitrary large velocities, so that the distance they can travel in a single step is unbounded" (1980: 145).

More importantly still, we argue that Gandy's characterization does not even cover all cases of computation that are in accord with the principle of local causation

and his two overarching physical presuppositions (an upper bound on the speed of propagation of effects and signals, and a lower bound on the dimensions of the assembly's components). We consider discrete mechanical systems that infringe Thesis M in the next section, but we begin with some general considerations about physical computation.

4. Is the physical world computable?

The issue of whether every aspect of the physical world is Turing computable was raised by several authors in the 1960s and 1970s, and the topic rose to prominence in the mid-1980s. In 1985, Wolfram formulated a thesis that he described as "a physical form of the Church-Turing hypothesis": this says that the universal Turing machine can simulate any physical system (1985: 735, 738). In the same year David Deutsch (who laid the foundations of quantum computation) formulated a principle that he also called "the physical version of the Church-Turing principle" (Deutsch 1985: 99). Other formulations were advanced by Earman (1986), Pour-El and Richards (1989), Pitowsky (1990), and Blum et al. (1998).

In the 1990s Copeland coined the term "hypercomputer" for any system—notional or real, natural or artefactual—that computes functions, or numbers, that the universal Turing machine cannot compute (Copeland and Proudfoot 1999, Copeland 2002). A processing system—either a computing system, or a system of some other sort—is said to be "hypercomputational" if the information-processing that it performs cannot be done by the universal Turing machine (Copeland 2000). Scott Aaronson has suggested (in correspondence) that the physical Church-Turing thesis be called simply the *anti-hypercomputation thesis*. The term "physical Church-Turing thesis" is far from ideal, since the Church-Turing thesis as Turing and Church put it forward concerned only the scope and limits of human computation (Copeland 1996, 2017); however, we will continue to use the term here (since many do use it).

We use the term *physical* to refer to systems whose operations are in accord with the actual laws of nature. These include not only actually existing systems but also idealized physical systems (systems that operate in some idealized conditions) and physically possible systems that do not actually exist, but that *could* exist, or did exist (e.g. in the universe's first moments), or will exist. Of course, there is no consensus about exactly what counts as an idealized or possible physical system, but this is not our concern here.

Gualtiero Piccinini distinguishes between what he calls "bold" and the "modest" versions of the physical Church-Turing thesis (2011, 2015). (The distinction applies equally to versions of the anti-hypercomputation thesis.) Bold versions concern physical systems and processes in general, while modest versions are about systems that themselves compute and processes that themselves qualify as computation. Wolfram's thesis is an example of a bold version:

> **Wolfram's bold physical Church-Turing thesis**: "[U]niversal computers are as powerful in their computational capacities as any physically realizable system can be, so that they can simulate any physical system." (Wolfram 1985: 738)

The formulations of Deutsch and others are also bold: their formulations concern physical systems in general and not just computing systems. (Piccinini emphasizes, though, that the bold versions proposed by different writers are often "logically independent of one another", and exhibit "lack of confluence" (2011: 747-748).) Modest versions of the physical Church-Turing thesis, on the other hand, concern physical systems that themselves compute, and assert that the computational power of *any physical computer* is bounded by Turing computability. Gandy's thesis is an example. His Thesis M is about *calculating machines* and his talk about functions that are calculated (or computed) by machines—DDMAs—implies that the mediating processes are calculations (computations).

Nevertheless, Gandy's result implies a bold version: since DDMAs are physical systems, Gandy proved that the behaviour of a certain broad class of physical systems is bounded by Turing computability. First, though, we will discuss the modest thesis. Is it true? Given that Gandy proved that Turing computability is an upper bound on the computational powers of DDMAs, the pertinent question is whether computing systems other than DDMAs are able to compute functions that are not Turing computable.

There have been several attempts to cook up constructions of highly idealized physical machines that compute functions that no Turing machine is able to compute. Perhaps the most interesting ones have been of "supertask" machines—machines that complete infinitely many computational steps in a *finite* span of time. Among such machines we find accelerating machines (Copeland 1998a, 2002b, Copeland and

Shagrir 2011), shrinking machines (Davies 2001), and relativistic machines (Pitowsky 1990, Hogarth 1994, Andréka et al. *this volume*).

Relativistic machines operate in spacetime structures having the property that the entire endless lifetime of one machine is included in the finite chronological past of another machine (called "the observer"): thus the first machine could carry out an infinite computation, such as calculating every digit of π, in what is from the observer's point of view a finite timespan, say one hour. (Such structures, sometimes called Malament-Hogarth spacetimes, are in accord with Einstein's General Theory of Relativity.)

A relativistic machine *RM* consists of a pair of communicating Turing machines T_A and T_B: T_A, the observer, is in motion relative to T_B, a universal machine. *RM* is able to "compute" the halting function. When the input *(m,n)*—asking whether the m^{th} Turing machine (in some enumeration of the Turing machines) halts or not when started on input *n*—enters T_A, T_A first prints 0 (meaning "never halts") in its designated output cell and then transmits *(m,n)* to T_B. T_B simulates the computation performed by the m^{th} Turing machine when started on input *n* and sends a signal back to T_A if and only if the simulation terminates. If T_A receives a signal from T_B, it deletes the 0 it previously wrote in its output cell and writes 1 there instead (meaning "halts"). After one hour, T_A's output cell shows 1 if the m^{th} Turing machine halts on input *n* and shows 0 if the m^{th} machine does not halt on *n*.

RM is of interest since arguably it complies with Gandy's principles. *RM* is discrete, since it consists of two standard digital computers in communication; and (as a relativistic machine) the speed of signal propagation in *RM* is bounded by the speed of light. Nonetheless, *RM* cannot be a Gandy machine if it computes a function that no Gandy machine is able to computes. So what is going on? Our answer is that *RM* violates an *implicit assumption* that underlies Gandy's Principle I (Copeland and Shagrir 2007). Principle I requires that the process can be described as a sequence S_0, $F(S_0)$, $F(F(S_0))$,… (where S_0 is the initial state and F is the state-transition function). But it is also assumed that the configuration of each stage $\alpha + 1$, described by S_{i+1}, is to be *uniquely determined* by the configuration of *the* previous stage, α, described by S_i (i.e. that $S_{i+1} = F(S_i)$). We will call this the assumption of *Gandy determinism*. However, this assumption is not necessarily satisfied by *RM*. Consider the end-stage of T_A: if T_A receives a signal from T_B, then its subsequent behavior is Gandy-

deterministic; but if it receives no signal from T_B, its behavior is no longer Gandy deterministic. To count as Gandy-deterministic, the end-stage of T_A-halting-on-0 should be determined, in part, by the no-signal message of *the* last stage of T_B. However, T_B, a non-halting Turing machine, does not have a last stage: there is no stage of T_B that is the one coming just before the end-stage of T_A-halting-on-0 (since after each stage of T_B, there are infinitely many others at which no signal is sent to T_A). Thus the stage of T_A-halting-on-0 is not Gandy-deterministic.

This implicit assumption is the weak point in Gandy's argument, since not every deterministic assembly need be Gandy-deterministic. Moreover there is an extremely reasonable account of determinism according to which *RM is* deterministic. It is deterministic in that the end-stage of T_A-halting-on-0 is uniquely determined by the initial stage of the machine. This is because the end-stage of T_A-halting-on-0 is a *limit* of previous stages of T_B (and T_A), of which the relevant feature is their not sending a signal to T_A. This sense of determinism is in good accord with physical usage where a system or machine is said to be deterministic if it obeys laws that invoke no random or stochastic elements. T_A's halting on 0 is completely determined by the fact that it initially wrote 0 in its designated output cell and the fact that at no stage of the computation was a signal sent by T_B.

RM is not a Gandy machine but it is a DDMA (although not a Gandy-deterministic DDMA). Is it a counter-example to the modest thesis? This depends on whether the machine is *physical* and on whether it really *computes* the halting function.

Is *RM* physical? Németi and his colleagues provide the most physically realistic construction, locating machines like *RM* in setups that include huge slow rotating Kerr black holes (Andréka et al. *this volume*) and emphasizing that the computation is physical in the sense that "the principles of quantum mechanics are not violated" and *RM* is "not in conflict with presently accepted scientific principles" (Andréka, Németi and Németi 2009: 501). They suggest that humans might "even build" their relativistic computer "sometime in the future" (Andréka, Németi and Németi 2009: 501). Naturally all this is controversial. Earman and Norton (1993), Aaronson (2005), Piccinini (2011), and others, argue that this relativistic physical setup faces serious problems: however, Németi and his colleagues reply resourcefully

to these objections (Etesi and Németi (2002), Németi and Dávid (2006), Andréka et al. (2009) and Andréka et al. (*this volume*); see also Shagrir and Pitowsky (2003)).

Does *RM compute* the halting function? The answer depends on what is included under the heading *physical computation*. We cannot possibly cover here the array of differing accounts of physical computation found in the current literature. But we can say that *RM* computes in the senses of "compute" staked out by several of these accounts: the semantic account (Shagrir 2006, Sprevak 2010), the mechanistic account (Milkowski 2013, Fresco 2014, Piccinini 2015), the causal account (Chalmers 2011), and the BCC (broad conception of computation) account (Copeland 1997a: 695). According to all these accounts, *RM* counterexamples the modest thesis *if RM* is physical. However, *RM* does *not* compute if computation is construed as the execution of an algorithm in the classical sense. The classical notion of an algorithm does not accommodate the limit stages found in relativistic computation (although it does accommodate all sorts of nondeterministic processes, e.g. probabilistic processes).

We conclude that Gandy's principles do not provide a general and comprehensive analysis of machine computation. We do not wish to downplay the contribution that his analysis has made to the current understanding of machine computation; but it is important to realize that his analysis is limited in its scope. In fact, its scope is more limited than is suggested by Gandy's own exclusion of analogue machines and some types of discrete Newtonian machines: his analysis does not even cover all instances of non-hypercomputational discrete physical computation. For instance, his Principle I does not directly apply to probabilistic algorithms and asynchronous algorithms (Gurevich 2012, Copeland and Shagrir 2007).

We turn now to the bold thesis, which says in effect that the behaviour of every physical system can be *simulated* (to any required degree of precision) by a Turing machine. Speculation that there may be physical processes whose behaviour cannot be calculated by the universal Turing machine stretches back over several decades (for a review see Copeland 2002a). Early papers by Scarpellini (1963), Komar (1964) and Kreisel (1965, 1967) made this point. Georg Kreisel stated "There is no evidence that even present day quantum theory is a mechanistic, i.e. recursive theory in the sense that a recursively described system has recursive behaviour" (1967: 270). More concretely, Marian Pour-El and Ian Richards (1981) showed that

the familiar three-dimensional wave equation produces non-Turing-computable output sequences for some Turing computable input sequences. But their result is at the mathematical level: it is an open question whether the requisite input sequences can obtain physically. *RM* (if physical) provides another counterexample to the bold theses (since the bold thesis implies the modest).

To summarize the discussion so far: the bold thesis is clearly an empirical hypothesis, and at the present stage of physical enquiry it is unknown whether this hypothesis is true. However, it can at least be said that to date there is no empirical evidence against the hypothesis (so far as we know). The modest thesis also seems to be an empirical hypothesis, although here matters are more complex, since a conceptual issue also bears on the truth or falsity of the thesis—the issue of what counts as physical computation. As with the bold thesis, it is currently unknown whether the modest thesis is true or false.

Next we introduce a new, stronger, form of the physical Church-Turing thesis and examine some recent work on undecidability in physics. We call this new form the "super-bold" physical Church-Turing thesis. Unlike the bold thesis, it concerns not only the ability of the universal Turing machine to simulate the behaviour of physical systems (to any required degree of precision) thesis but also concerns further physical questions about this behaviour. Examples are decidability questions such as: "Is the solar system stable?" and "Is the motion of a given system, in a known initial state, periodic?" (Pitowsky 1996).

> **The super-bold physical Church-Turing thesis**: *Every aspect of the behaviour of any physical system is Turing computable (to any desired degree of accuracy).*

In 1986 Robert Geroch and James Hartle argued that *un*decidable physical theories "should be no more unsettling to physics than has the existence of well-posed problems unsolvable by any algorithm have been to mathematics"; and they suggested that such theories may be "forced upon us" in the quantum domain (Geroch and Hartle 1986: 534, 549). Arthur Komar raised "the issue of the macroscopic distinguishability of quantum states" in 1964, asserting that there is no effective procedure "for determining whether two arbitrarily given physical states can be superposed to show interference effects" (Komar 1964: 543-544). More recently Jens Eisert, Markus Müller and Christian Gogolin showed that "the very natural physical problem of determining whether certain outcome sequences cannot occur in repeated

quantum measurements is undecidable, even though the same problem for classical measurements is readily decidable" (Eisert, Müller and Gogolin 2012: 260501-1). This is an example of a problem that refers unboundedly to the future but not to any specific time (as in Itamar Pitowsky's examples mentioned earlier). Eisert, Müller and Gogolin suggest that "a plethora of problems" in quantum many-body physics and quantum computing may be undecidable (2012: 260501-1 - 260501-4).

Dramatically, a 2015 *Nature* article by Toby Cubitt, David Perez-Garcia, and Michael Wolf outlined their proof that "the spectral gap problem is algorithmically undecidable: there cannot exist any algorithm that, given a description of the local interactions, determines whether the resultant model is gapped or gapless" (Cubitt et al. 2015: 207). Cubitt describes this as the "first undecidability result for a major physics problem that people would really try to solve" (in Castelvecchi 2015).

The spectral gap, an important determinant of a material's properties, refers to the energy spectrum immediately above the ground energy level of a quantum many-body system (assuming that a well-defined least energy level of the system exists): the system is said to be gapless if this spectrum is continuous and gapped if there is a well-defined next least energy level. The spectral gap problem for a quantum many-body system is the problem of determining whether the system is gapped or gapless, given the finite matrices describing the local interactions of the system.

In their proof Cubitt et al. encode the halting problem in the spectral gap problem, so showing that the latter is at least as hard as the former. The proof involves an infinite family of 2-dimensional lattices of atoms; but they point out that their result also applies to finite systems whose size increases: "Not only can the lattice size at which the system switches from gapless to gapped be arbitrarily large, the threshold at which this transition occurs is uncomputable" (Cubitt et al. 2015: 210-211). Their proof offers an interesting countermodel to the super-bold thesis, involving a physically relevant example of a finite system of increasing size such that there exists no Turing computable procedure for extrapolating the system's future behavior from (complete descriptions of) its current and past states. (For discussion of such systems, see Geroch and Hartle 1986 and Copeland 2002, 2004.)

It is debatable whether any of these quantum models corresponds to real-world quantum systems. The Komar model involves a system with an infinite number of degrees of freedom; and Cubitt et al. admit that the model invoked in their proof is

highly artificial, saying "Whether the results can be extended to more natural models is yet to be determined" (Cubitt et al. 2015: 211). There is also the question of whether the spectral gap problem becomes computable when only local Hilbert spaces of realistically low dimensionality are considered. Nevertheless, these results are certainly suggestive. The super-bold thesis cannot be taken for granted—even in a finite quantum universe.

We turn next to Penrose's speculations concerning physical uncomputability.

5. Penrose's thesis: uncomputability and the brain

Penrose's thesis is the claim that the action of the brain is hypercomputational (Penrose 2013: xxxiii). Penrose holds that the brain's uncomputability is key to explaining the phenomenon of consciousness (Penrose 1989, 1990, 1994, Hameroff and Penrose 2014). According to Penrose, the brain's hypercomputational action, and the role this plays in generating conscious experience, will not be fully understood until the advent of what he calls the New Theory in physics: he says that "hypercomputational actions" in the brain are the "non-computable effects of [the] New Theory" (Penrose 2013: xxxiii). This "presumed New Theory", he says, goes "beyond current quantum mechanics": it is "presently unknown in detail" and involves "hitherto undiscovered laws" (2013: xxxii, xxxiii).

Penrose's argument for his thesis is based on Gödel's incompleteness theorems, which he "regard[s] as providing a strong case for human understanding being something essentially non-computable"—understanding being "one manifestation of human consciousness" (Penrose 2013: xxviii, 2011: 347). This general line of argument, made famous in an article by the philosopher John Lucas (Lucas 1961), is often called the "Gödel Argument", although in fact it was anticipated by Emil Post as early as 1921 (Post 1965: 417). Penrose calls it the "Gödel-Turing Argument" (e.g. in his 2011); and Turing himself dubbed it the "Mathematical Objection" (1950: 450), giving the following elegant summary of it:

> Recently the theorem of Gödel and related results ... have shown that if one tries to use machines for such purposes as determining the truth or falsity of mathematical theorems and one is not willing to tolerate an occasional wrong result, then any given machine will in some cases be unable to give an answer at all. On the other hand the human intelligence seems to be able to find methods of ever-

> increasing power for dealing with such problems "transcending" the methods available to machines. (Turing 1948: 410-11.)

Turing by no means endorsed the "Gödel-Turing Argument". His subtle objection to it, involving what we call his "multi-machine theory" of mentality, is described in Copeland and Shagrir (2013)—and is very different from the objection that Penrose imputes to Turing, in our view mistakenly (e.g. in his 1997: 112). We shall return briefly to Turing's views below.

Gödel's view, as he expressed it in his 1951 Gibbs lecture, was that the incompleteness results establish a *disjunction*: *either* "there exist absolutely unsolvable diophantine problems" (where, Gödel explained, "the epithet 'absolutely' means that they would be undecidable, not just within some particular axiomatic system, but by *any* mathematical proof the human mind can conceive"), *or else* "the human mind … infinitely surpasses the powers of any finite machine" (Gödel 1951, p. 310). (For a fuller study of Gödel's views, see Copeland and Shagrir 2013).

Later, at the beginning of the 1970s, Gödel in effect recast this disjunction into an implication:

> If my result [incompleteness] is taken together with the rationalistic attitude which Hilbert had and which was not refuted by my results, then [we can infer] the sharp result that mind is not mechanical. This is so, because, if the mind were a machine, there would, contrary to this rationalistic attitude, exist number-theoretic questions undecidable for the human mind. (Gödel in Wang 1996: 186-187)

What Gödel called Hilbert's "rationalistic attitude" was summed up in the latter's celebrated remark that "in mathematics there is no *ignorabimus*"—there is no mathematical question that in principle the mind is incapable of settling (Hilbert 1902: 445).

Gödel's position, then, was that his incompleteness results do *not* entail that the mind is not mechanical; but, if these are coupled with the rationalistic attitude that there are *no* absolutely undecidable problems—an attitude that, he emphasized, "remains entirely untouched" by his negative results (Gödel 193?, p.164)—then it does indeed follow that the mind is not mechanical. In a note written in 1963 (Fig. 1) Gödel explained where he sat in this debate (at any rate at that time). Referring to his

1951 disjunction he said: "I believe, on ph[ilosophical] grounds, that the sec[ond] alternative is more probable & hope to make this evident".[2]

PLACE FIGURE 1 NEAR HERE

Caption: Extract from Gödel's draft letter of 1963, addressed to TIME Inc.[3]

Credit: Firestone Library, Rare Books and Special Collections

Clearly the success of the Gödel Argument turns on whether this "rationalistic attitude" could ever be established to be correct—i.e. whether it could ever be established that there are no absolutely undecidable problems. It is, to be sure, difficult to see how this could ever be done. But, in any case, Gödel's "sharp result" is undercut by Turing's rebuttal of the Mathematical Objection (see Copeland and Shagrir 2013). Moreover, numerous other objections have been raised to the Gödel Argument, and to the detailed formulation of it endorsed by Penrose (see for example Penrose 1990 and the commentaries that follow). Rather than attempting to survey these many objections here, we will focus on what seems to us to be the absolutely central difficulty with Penrose's argument, namely that the argument appears to reduce to absurdity (Copeland 1998b, Copeland and Proudfoot 2007).

The reductio ad absurdum is this. Let us suppose Penrose's argument does successfully establish that (as he puts it) human mathematicians do not use a knowably sound Turing-machine algorithm in order to ascertain mathematical truth. If so, then his argument shows with equal success that human mathematicians, in ascertaining mathematical truth, do not use any knowably sound procedure that is capable of being executed by an *oracle machine*. Turing's oracle machines (or *o*-machines) are the result of equipping a universal Turing machine with at least one additional basic operation that no Turing machine proper can simulate (Turing 1939). Turing called these new basic operations "oracles", saying that oracles work by "some unspecified means" (1939: 156).

As Turing explained, oracle machines form a hierarchy that extends ever upwards. Let the *first-order o*-machines be those whose oracle produces the values of the Turing-machine halting function $H(x,y)$. The *second-order o*-machines are those with an oracle that can say whether or not any given first-order *o*-machine eventually

[2] See van Atten and Kennedy (2003) for a discussion of Gödel's 1963 note.
[3] Thanks to Juliette Kennedy for assistance in locating this note.

halts if set in motion with such-and-such a number inscribed on its tape; and so on for third-order and in general α-order *o*-machines. Penrose's argument was originally marketed as showing that human understanding does not consist in any process that a Turing machine can execute (see e.g. Penrose 1994, ch. 2); but his argument is so powerful that it equally supports the conclusion that human understanding does not consist in any process that the richly hypercomputational oracle machines can execute. (This applies even to the "cautious oracles" that Penrose introduces in his 2016.) Penrose's argument moves relentlessly up through the orders, stopping nowhere.

Penrose noted this difficulty in his 1994 book *Shadows of the Mind* (p. 380). He also suggested a way out:

> [I]t need not be the case that human mathematical understanding is in principle as powerful as *any* oracle machine at all. ... Thus, we need not necessarily conclude that the physical laws that we seek reach, in principle, beyond every computable level of oracle machine (or even reach the first order). We need only seek something that is not equivalent to *any* specific oracle machine. (1994: 381.)

What does Penrose mean here? It is customary in recursion theory to say that problems of equal "hardness" are of the same *degree*: problems that are solvable by Turing machines are said to be of degree 0. Penrose seems to be suggesting that physical laws occupy a position *in between* degree 0 and degree 1, the degree of problems that are solvable by a first-order oracle machine but not by Turing machine. It is indeed known that there are degrees between 0 and 1 (Friedberg 1957, Sacks 1964) and this seems to make sense of what Penrose is suggesting: for some degree between 0 and 1, the "physics of mind" is exactly that hard. This is certainly a coherent position—and for all that anyone presently knows, it may in fact be true.

However, this suggestion does not prevent the reductio ad absurdum that we are discussing (Copeland 1998b). Let *i* (for "intermediate") be a degree between 0 and 1 and let *I* be the class of *o*-machines that are able to solve problems of degree *i* (and no harder problems). Do mathematicians use, in ascertaining mathematical truth, a knowably sound procedure able to be executed by a machine in *I*? Not if Penrose's argument is sound, since it applies equally to the *o*-machines in *I*. To borrow a phrase of Penrose's (from his 2013: xxxiv), the Gödel Argument involves a "never-ending

capability of being able to 'stand back' and contemplate whatever structure had been considered previously": whatever structure—whatever physical system—is contemplated, the argument deems it not to be the mind.

In his more recent work Penrose does not repeat the suggestion just discussed. But nor does he offer any way of avoiding the reductio ad absurdum that he noted in his 1994 book. Commenting on the fact that, no matter what device *D* is specified, the Gödel Argument entails that the mind is more powerful than *D*, Penrose says only that the mind is "something very mysterious" and that its theory must involve "something very subtle" (Penrose 1996, sect. 13.2, Penrose 2013: xxxiv). John Lucas was happy to conclude from the Gödel Argument that "no scientific enquiry can ever exhaust the ... human mind" (Lucas 1961: 127), and Gödel thought that the brain must be "a computing machine connected with a spirit" (Gödel in Wang 1996, p. 193). Unlike Gödel and Lucas, Penrose seems to think that there must be a fully physical account of consciousness, but he has failed to make it clear what physical conception of consciousness can possibly remain for one who endorses the Gödel Argument.

It is a pity that Penrose chose to support his thesis by means of the Gödel Argument, since the argument is ultimately a distraction and moreover tends to mask the fact that Penrose's thesis is—like the various forms of the physical Church-Turing thesis considered above—a thoroughly empirical thesis. It is a serious hypothesis that, far from requiring a radical New Theory, might even be consistent with current quantum mechanics, as the undecidability of the spectral gap problem tends to indicate. There is, so far as we aware, not a shred of empirical evidence for Penrose's thesis, but this situation might change in the future. One can only keep an open mind.

We conclude with a comment on the relationship between Penrose's view of the brain and Turing's. Penrose says: "It seems likely that he [Turing] viewed physical action in general—which would include the action of a human brain—to be always reducible to some kind of Turing-machine action" (Penrose 1994: 21). Penrose even named this claim *Turing's thesis*. Yet Turing never endorsed this thesis and was aware that it might be false. Turing was in fact an important forerunner of the modern debate concerning the possibility of uncomputability in physics and uncomputability in the action of the human brain (as was first pointed out in Copeland 1999 and Copeland and Proudfoot 1999). In a 1951 lecture on BBC radio Turing suggested that it may not be possible for a computer to simulate the human brain because of the brain's

quantum-mechanical nature (Turing 1951, Copeland 1999: 448, 451-2). Far from subscribing to what Penrose called Turing's thesis, Turing in this lecture contemplated the possibility that the physics of the brain might be uncomputable. (Even Andrew Hodges, who used to maintain that Turing claimed "that the action of the brain must be computable" (Hodges 2003: 51), now seems to have accepted that his previous view of Turing was wrong (Hodges 2012).)

6. Summary

We have discussed a number of theses concerning the relationship between physics and computation. We began with the thesis that the physical universe is a computer (Zuse's thesis) and moved on to the thesis that the behaviour of all discrete deterministic mechanical assemblies is Turing computable (Gandy's thesis) and then the more general physical Church-Turing thesis (which is also known as the "anti-hypercomputation thesis"). We distinguished three versions of the physical Church-Turing thesis: the modest, the bold, and the super-bold versions. We ended with the thesis that some actions of a specific physical system—the human brain—are not Turing computable (Penrose's thesis).

These are all exciting hypotheses, but we conclude that none of them is empirically validated. Zuse's thesis has not yet proved sufficiently useful in fundamental physics for us to wish to embrace its racy ontological commitments. Gandy's thesis turns out to be confined to discrete mechanical assemblies that are deterministic in a very specific sense, which we dub Gandy deterministic, and fails to apply to assemblies that are deterministic in other familiar senses. Whether the behaviour of those deterministic assemblies also is bounded by Turing computability remains an open question. In particular, Gandy's thesis fails to rule out relativistic computation. Other physical versions of the Church-Turing thesis—bold and super-bold—are more general, but their validity is also questionable. Penrose's thesis, too, emerges as an interesting speculation for which evidence is currently wanting.

References

Aaronson, S. "Guest column: NP-complete problems and physical reality." *ACM Sigact News* 36.1 (2005): 30-52.

Bekenstein, J. D. (2007) “Information in the Holographic Universe.” *Scientific American* 17 (April), 66–73.

Blum, L., Cucker, F., Shub, M. and Smale, S. 1998: *Complexity and Real Computation*. Springer.

Castelvecchi, D. "Paradox at the heart of mathematics makes physics problem unanswerable." *Nature* 528 (2015): 207.

Copeland, B. J. 1997a. 'The Broad Conception of Computation'. *American Behavioral Scientist*, 40: 690-716.

— 1997b, 2017: The Church-Turing Thesis. *The Stanford Encyclopedia of Philosophy*, <http://plato.stanford.edu/entries/church-turing/ >.

— (1998a), Even Turing Machines Can Compute Uncomputable Functions, in C. Calude, J. Casti and M. Dinneen, eds., *Unconventional Models of Computation*, Springer.

— (1998b). Turing’s O-Machines, Penrose, Searle, and the Brain. *Analysis*, 58: 128-138.

— (1999). A Lecture and Two Radio Broadcasts on Machine Intelligence by Alan Turing. In Furukawa, K., Michie, D., Muggleton, S. 1999 (eds) *Machine Intelligence 15* Oxford University Press, 445-476.

— 2000. Narrow Versus Wide Mechanism, *Journal of Philosophy*, 96: 5-32, repr. in Scheutz, M. (ed.) 2002. *Computationalism*, MIT Press.

— 2002a: Hypercomputation. *Minds and Machines* 12: 461-502.

— 2004. Computable Numbers: A Guide. In Copeland ed. 2004: 5-57.

— ed. 2004. *The Essential Turing*. Oxford University Press.

—, Proudfoot, D. 1999. Alan Turing's Forgotten Ideas in Computer Science, *Scientific American*, 280 (4): 99-103.

— Proudfoot, D. 2007. Artificial Intelligence: History, Foundations, and

Philosophical Issues. In Thagard, P. 2007 (ed.) *Handbook of the Philosophy of Psychology and Cognitive Science*. Elsevier.

—, Shagrir, O. 2007: Physical Computation: How General are Gandy's Principles for Mechanisms? *Minds and Machines* 17: 217-231.

—, O. Shagrir. 2011. Do accelerating Turing machines compute the uncomputable? *Minds and Machines* 21:221–239.

—, Shagrir, O. (2013). "Turing versus Gödel on Computability and the Mind." In Copeland, B. J., Posy, C. and Shagrir, O. (eds.) (2013: 1-33). *Computability* MIT Press.

—, Sprevak, M., Shagrir, O. 2017. Is the Whole Universe a Computer? In Copeland, B. J. et al. *The Turing Guide* (Oxford University Press), 445-462.

Cubitt, T. S., D. Perez-Garcia, and M. M. Wolf. "Undecidability of the spectral gap." *Nature* 528.7581 (2015): 207-211.

Davies, B. E. (2001). "Building Infinite Machines." *British Journal for the Philosophy of Science* 52:671–682.

Dennett, D. C. (1991). "Real patterns". *The Journal of Philosophy* 88: 27-51.

Deutsch, D. 2003. 'It from qubit'. In J. Barrow, P. Davies, and C. Harper, eds, *Science and Ultimate Reality*. Cambridge University Press, 90–102.

Earman, J. 1986: A Primer on Determinism. Reidel.

Eisert, J., M. P. Müller, and C. Gogolin. "Quantum measurement occurrence is undecidable." *Physical review letters* 108.26 (2012): 260501.

Fresco, N. (2014). *Physical Computation and Cognitive Science* Springer.

Gardner, M. (1970). 'Mathematical Games – The fantastic combinations of John Conway's new solitaire game "life"'. *Scientific American* 223 (October): 120–123.

Geroch, R., Hartle, J.B. (1986), 'Computability and Physical Theories', *Foundations of Physics* 16, pp. 533–550.

Gödel, K. 193?. Undecidable Diophantine propositions. In Gödel, *Collected Works III,* 164–175.

— 1951.Some basic theorems on the foundations of mathematics and their implications. In Gödel, *Collected Works III,* 304–323.

Gurevich, Y. 2012: What is an Algorithm? In *SOFSEM: Theory and Practice of Computer Science* (eds. M. Bieliková, G. Friedrich, G. Gottlob, S. Katzenbeisser and G. Turán), *Springer LNCS* 7147: 31-42.

Hameroff S, Penrose R. 2014 Consciousness in the Universe: A Review of the "Orch OR" Theory. *Physics of Life Reviews*. 11: 39-78.

Hilbert, D. 1902. Mathematical problems: Lecture delivered before the International Congress of Mathematicians at Paris in 1900. *Bulletin of the American Mathematical Society* 8:437–479.

Hodges, A. 2003. What Would Alan Turing Have Done After 1954? In C. Teuscher (ed.) 2003. *Alan Turing*, Springer.

— 2012. Beyond Turing's Machines, *Science*, 336 (13 April 2012), 163-4.

Komar, A. (1964), 'Undecidability of Macroscopically Distinguishable States in Quantum Field Theory', *Physical Review*, second series, 133B, pp. 542–544.

Kreisel, G. (1965) 'Mathematical Logic', in T.L. Saaty, ed., Lectures on Modern Mathematics, Vol. 3, Wiley.

— (1967), 'Mathematical Logic: What Has it Done For the Philosophy of Mathematics?', in R. Schoenman, ed., Bertrand Russell: Philosopher of the Century, Allen and Unwin.

Lloyd, S. (2006) 'A theory of quantum gravity based on quantum computation', https://arxiv.org/abs/quant-ph/0501135.

Lucas, J. R. 1961. Minds, Machines and Gödel, *Philosophy*, 36: 112-127.

Mach, E. 1911. *The History and Root of the Principle of Conservation of Energy*. Open Court.

Miłkowski, M. (2013). *Explaining the Computational Mind* MIT Press.

Penrose, R. 1989 *The Emperor's New Mind*. Oxford University Press.

— 1990 Précis of *The Emperor's New Mind*, 13: 643-655, 692-705.

— 1994 *Shadows of the Mind*. Oxford University Press.

— 1996 'Beyond the Doubting of a Shadow' *Psyche*, 2 <http://psyche.cs.monash.edu.au/>.

— 1997. *The Large, the Small and the Human Mind*, Cambridge University Press.

— 2011. Gödel, the Mind, and the Laws of Physics. In Baaz, M., Papadimitriou, C. H., Scott, D. S., Putnam, H., and Harper, C. L. (eds) *Kurt Gödel and the Foundations of Mathematics*. Cambridge University Press.

— 2013. Foreword to Zenil 2013.

— 2016. On Attempting to Model the Mathematical Mind. In Cooper, S. B., Hodges, A. eds, 2016, *The Once and Future Turing*. Cambridge University Press.

Pitowsky, I. 1996 "Laplace's demon consults an oracle: The computational complexity of prediction." *Studies In History and Philosophy of Science Part B* 27.2: 161-180.

Poincaré, H. (1902). *Science and Hypothesis*. Repr. Dover, 1952.

Pour-El, M. B., Richards, I. J. 1981: The Wave Equation with Computable Initial Data such that its Unique Solution is not Computable. *Advances in Mathematics* 39: 215-239.

— (1989). *Computability in Analysis and Physics* Springer.

Russell, B. 1927. *The Analysis of Matter*. Kegan Paul, Trench, Trubner.

Sacks, G. E. 1964. The Recursively Enumerable Degrees are Dense, *Annals of Mathematics*, 2nd series, 80: 300-312.

Scarpellini, B. (1963), 'Zwei Unentscheitbare Probleme der Analysis', *Zeitschrift für mathematische Logik und Grundlagen der Mathematik* 9: 265–289. English translation in *Minds and Machines* 12 (2002): 461-502.

Schmidhuber, J. 2013. 'The fastest way of computing all universes'. In Zenil ed. 2013.

Shagrir, O. (2006). "Why We View the Brain as A Computer." *Synthese* 153:393-416.

Sieg, W. 2002: Calculations by Man and Machine: Conceptual Analysis. In *Reflections on the Foundations of Mathematics* (eds., W. Sieg, R. Sommer and C. L. Talcott). Association for Symbolic Logic, pp. 396-415.

—, Byrnes, J. 1999: An Abstract Model for Parallel Computations: Gandy's Thesis.

Monist 82: 150-164.

't Hooft, G. 2002. 'Looking at life with Gerardus 't Hooft'. *Plus Magazine* (January). http://plus.maths.org/issue18/features/thooft/

Turing, A. M. 1948. Intelligent Machinery. In Copeland, *The Essential Turing*, 410–432.

— Computing Machinery and Intelligence. In *The Essential Turing*, 441–464.

— 1951. Can Digital Computers Think? In *The Essential Turing*, 476–486.

van Atten, M., Kennedy, J. 2003. On the Philosophical Development of Kurt Gödel, *Bulletin of Symbolic Logic*, 9: 425-476.

Wallace, D. 2003. "Everett and Structure". *Studies in History and Philosophy of Science Part B*, 34: 87-105.

Wang, H. 1996. *A Logical Journey*. MIT Press.

Wheeler, J. A. 2006. Interview on 'The anthropic universe'. *The Science Show*. 18 February 2006. http://www.abc.net.au/radionational/programs/scienceshow/the-anthropic-universe/3302686

Wolfram, S. 1985: Undecidability and Intractability in Theoretical Physics. *Physical Review Letters* 54: 735–738.

Zenil, H. ed. 2013. *A Computable Universe*. World Scientific.