Physical Constraints on Hypercomputation

Paul Cockshott [a] Lewis Mackenzie [b] Greg Michaelson [c]

[a] *Department of Computing Science, University of Glasgow, 17 Lilybank Gds, Glasgow, G12 8QQ,* wpc@dcs.gla.ac.uk

[b] *Department of Computing Science, University of Glasgow, 17 Lilybank Gds, Glasgow, G12 8QQ,* lewis@dcs.gla.ac.uk

[c] *School of Mathematical and Computer Sciences, Heriot-Watt University, Riccarton, EH14 4AS,* greg@macs.hw.ac.uk

Abstract

"To infinity, and beyond!", Buzz Lightyear, Toy Story, Pixar, 1995.

Many attempts to transcend the fundamental limitations to computability implied by the Halting Problem for Turing Machines depend on the use of forms of *hypercomputation* that draw on notions of infinite or continuous, as opposed to bounded or discrete, computation. Thus, such schemes may include the deployment of actualised rather than potential infinities of physical resources, or of physical representations of real numbers to arbitrary precision. Here, we argue that such bases for hypercomputation are not materially realisable and so cannot constitute new forms of effective calculability.

1 Introduction

1.1 Turing Machines and Hypercomputation

The mathematical foundations of contemporary Computer Science are deeply rooted in Turing's 70 year old formulation of the Turing Machine (TM) as a basis for computation[1]. The unsolvability of the Halting Problem for TMs, as well as halting Hilbert's Programme, frames fundamental limitations to what can be achieved through what Church[2] termed "effective calculability", which Turing later characterised as finding values by "some purely mechanical process"[3]. Nonetheless, Turing had shown[1] that "it is possible to invent a single machine which can be used to compute any computable sequence." on the assumption that all computable sequences could be captured by TMs.

The theoretical ferment of the mid 1930's saw at least four accounts of effective calculability: Kleene's recursive functions[4] and Post's machines[5], as well as Turing's TMs and Church's λ-calculus[2]. In footnotes to the latter paper, Church notes that his results could be reformulated in Kleene's notation, and vice versa. He also notes that Godel had raised the question of the relationship between effective calculability and recursiveness in conversation with him, and that he himself had previously proposed the same question with regard to the λ-calculus.

In short order, all four systems were formally shown to be equivalent, that is any decision problem expressed in any one system could be reduced to an equivalent decision problem in any other. In turn this led to the posing of the Church-Turing Thesis, that all accounts of effective calculability are equivalent. Thus, the undecidability of the halting problem and results derived from it would apply to any Church-Turing system.

There have been many attempts to articulate new systems for computability [1] that are claimed to transcend the limitations of Church-Turing systems, termed for example *hypercomputing* or *super-Turing*. Copeland[6] provides a thorough summary.

In developing our critique[7,8] of Wegner and Eberbach's claims[9] that interaction machines, the π-calculus and the $-calculus transcend TMs, we were struck that many such attempts evade the canonical requirement of displaying a decision problem which is solvable in a new system but not solvable in a Church-Turing system. Instead, they often appeal to notions of infinite computation or continuous functions, which are at first sight appealing but of actually problematic physical status. It is such notions that we address here.

1.2 Infinities and Representations

As we shall see, proponents of hypercomputation may require the availability of infinite amounts of physical computing resource or of infinite amounts of time in which to perform computations. They may also require instantaneous communication between computing devices, that is they invoke infinitely small periods of time for information to pass between separated communicators.

Here we echo Aristotle[10] in making a strong decision between *potential* and *actualised* infinities. A potential infinity is one that corresponds to a process that could go on for ever but which can necessarily never be completed. In contrast, an actualised infinity must be concretely realised at some finite instance.

[1] Note that we somewhat carelessly use "computability", "effective calculability" and "effective computability" interchangeably

Unlike intuitionists or constructivists, do not oppose the idea of potential infinity as a purely mathematical concept; what we do dispute is the idea that actually achieved infinities can exist and be made use of in physical reality.

Furthermore, some proponents of hypercomputation attempt to deploy functions that compute over real numbers represented to infinite precision. This seems to repeat the above concerns as such representations require infinite resources and either infinite time to process them or no time to communicate them.

Note that we have no problem with infinite precision entities provided they have finite representations. For example:

```
e = exp 1
where
exp x = sum coeff (x,0)
sum f (x,n) = f (x,n)+sum f (x,n+1)
coeff (x,n)  = (power x n) / (fac n)
fac 0 = 1 | fac n = n*fac (n-1)
```

is a perfectly good finite representation of e which we can use in a range of mathematical applications. However, its *evaluation* to an expanded value requires an actualised infinity of resource. Curiously, Jeffrey[11], on which the above definition is based, comments “whatever this expression may mean”.

1.3 Are TM Tapes Infinite?

Many popular accounts of TM state baldly that a TM has an infinite tape, by implication an actualised infinity in Aristotle's sense. In fact Turing nowhere specified that the tape is infinite. On the contrary, Turing's implication that the TM tape is extended piecemeal as required, cell by cell. This differs strongly from the requirement that a Post machine should have an infinite bi-directional “sequence of spaces or boxes”[5].

We think that this is of fundamental importance in distinguishing the TM from recent attempts to push back the Church-Turing envelope by invoking actualised infinities of resources that are in principle available at the start of the computation. We would rather characterise a TM tape as of *bounded but unknown length*: if a TM terminates then it has carried out a finite number of steps and so can only have added a finite number of cells to the tape.

If a TM does not terminate, then there seem to be two possibilities. In a finite universe, material for new tape cells is eventually exhausted and the TM fails without completing. In an infinite universe, the TM just keeps on rolling along.

It is very interesting that Post machines, which require actualised infinities of tape from the start, are amongst the first systems to have been shown to satisfy the Church-Turing thesis. This suggests that appeals to actualised infinities as transcending TM limitations are ill-founded. Indeed, we speculate that a TM with an actually infinite tape is, in principle, no weaker than any other system requiring an actualised infinity of resource, as the π-calculus appears to do[8].

1.4 Operations, Rules and Physical Computation

A fundamental difference between Turing's formulation of the TM and other foundational systems, like recursive function theory and the λ-calculus, is that TMs are thoroughly grounded in physical machines. Turing[1] explicitly talks about his machines being able to do those things one might reasonably expect a human with paper and pencil to do. Indeed, his very simple description of the general idea of a TM is in terms of a real machine manipulating physically inscribed symbols.

In contrast, Church and Kleene present uninterpreted syntactic systems of symbols with formal rules for their manipulation. However, there is no explicit mechanism for the application of rules to symbol sequences. As we pointed out in [8], without mathematicians to apply rules, the symbols never change. While a TM embodies operations that are specific to the computation, expressions in λ-calculus or recursive function theory additionally require a general interpretation mechanism.

To see this, consider constructing a physical computation-specific TM using technology available in 1937. The transitions (quintuplets) can be realised as a combinatorial control circuit where the state and symbol inputs produce the new state, new symbol and direction outputs.

The tape can be realised as magnetic tape driven by a stepping transport mechanism with a left/right control line and a read/write head.

The TM is then structured as in Figure 1. The control sequence is to repeatedly:

- gate the state register and head contents into the control address lines;
- gate the control contents into the state register, head contents and shift left/right wire;
- if end of tape is detected, then operators splice in another length.

Note that this is not a *Universal* TM(UTM) but a computation-specific device as the control directly implements the quintuplets. However, a UTM could be

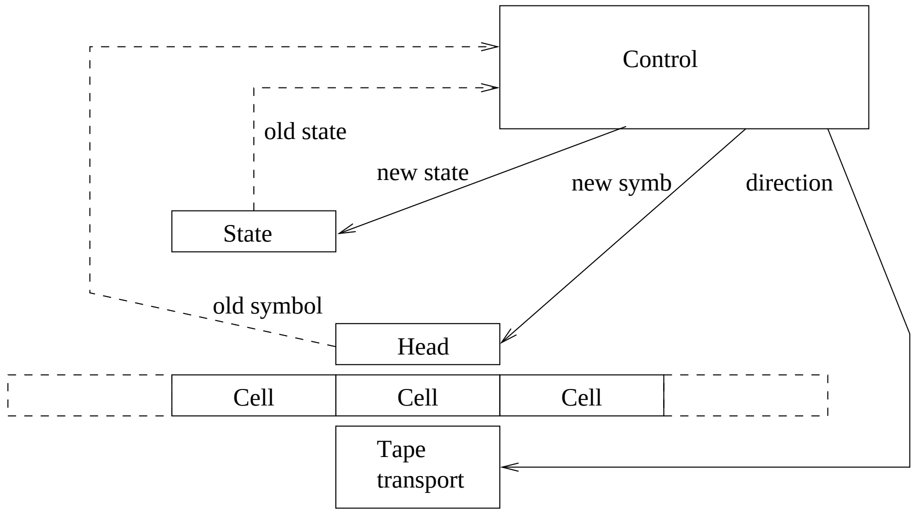

Fig. 1. Physical TM

loaded into the control.

In contrast, it is not possible to directly build a computation specific machine from an arbitrary λ-calculus or recursive function expression. Of course, the Church-Turing thesis tells us that it is always possible to compile an arbitrary construct in any computational notation into a concrete computation for physical realisation, say by abstract interpretation of the semantics.

So why are we so concerned with physical realism? While we are, of course, deeply interested in theoretical properties established formally, ultimately we are concerned with *effectivity*, that is with real systems performing physically embodied computation to solve problems in the real world. Any allegedly trans-Church-Turing system which is not physically realisable cannot be used to perform real computations to solve real problems. Contrariwise, we allege that any physically realisable system which can solve real problems will be Church-Turing.

For example, Wegner and Eberbach[9] make much of their assertions that the Internet and evolutionary computing systems cannot be truly modeled by TMs because they require interaction with an open ended environment where a TM's tape is not subject to external modifications. Noting that we dispute this in [8], nonetheless the Internet is made of real computers wired together, and machine learning systems run on real computers.

1.5 Physical Realism and Computation

Turing marks what Bachelard and Althusser[12] termed an Epistemological break in the history of the sciences. At once the problematic of Hilbertian rationalism is abandoned [13] and at the same time the Platonic view of mathematics is displaced by a materialist view.

As emphasised above, a key point about the Universal Computers proposed by Turing is that they are material apparatuses which operate by finite means. Turing assumes that the computable numbers are those that are computable by finite machines, and initially justifies this only by saying that the memory of a human computer is necessarily limited. By itself this is not entirely germane, since the human mathematician has paper as an aide memoir and the tape of the TM is explicitly introduced as an analogy with the squared paper of the mathematician.

> Computing is normally done by writing certain symbols on paper. We may suppose this paper is divided into squares like a child's arithmetic book. In elementary arithmetic the two-dimensional character of the paper is sometimes used. But such a use is always avoidable, and I think that it will be agreed that the two-dimensional character of paper is no essential of computation. I assume then that the computation is carried out on one-dimensional paper, i.e. on a tape divided into squares.([1] section 9)

However Turing is careful to construct his machine descriptions in such a way as to ensure that the machine operates entirely by finite means and uses no techniques that are physically implausible. His basic proposition remained that: "computable numbers may be described briefly as the real numbers whose expressions as a decimal are calculable by finite means.".

Turing thus rules out any consideration that computation by infinite means is a serious proposition. If infinite computation were to be allowed, then the limitations introduced by the TM would not apply. However, several proposals for super-Turing computation rest on the appeal of the infinite:

(1) Infinities of subdivision of time - these are proposed in the context of accelerated Turing machines[6] or infinite interleaving of messages [14].
(2) Infinite duration of computation [15] [16].
(3) Infinite spatial extension of computation[14], also proposed in [9], and criticised by us in [7].
(4) Infinite precision of measurement [17], but also implied in [14] and [18].

In the following sections, we will argue that all these ideas face insurmountable obstacles if our current understanding of the laws of physics are correct.

Note, however, that we do not here consider systems that depend for their efficacy on unspecified *oracles*, which somehow always find right answers. Turing[3], in introducing the idea, says that an oracle "cannot be a machine".

2 Accelerating TMs

Copeland[6] proposes the idea of accelerating TMs whose operation rate increases exponentially so that if the first operation were performed in a microsecond, the next would be done in $\frac{1}{2}\mu s$, the third in $\frac{1}{4}\mu s$, etc. The result would be that within a finite interval it would be able to perform an infinite number of steps. This obviously evades Turing's stipulation that computable numbers must be calculable by finite means, and at the same time evades all possibility of physical realisation. A computing machine must transfer information between its component parts in order to perform an operation. If the time for each operation is repeatedly halved. then one soon reaches the point at which signals traveling at the speed of light have insufficient time to propagate from one part to another within an operation step. Beyond this speed the machine could not function. Whilst in a hypothetical Newtonian universe without the constraint of a finite speed of light, this particular obstacle would be eliminated, one would immediately face another. In order for the Newtonian machine to compute infinitely fast, its (now presumably mechanical) components would have to move infinitely fast and thus require infinite energy. Given that the machine would be dissipative [19] the heat released would raise its temperature to an infinite extent causing it to disintegrate.

Hamkins[15] discusses what could be computed on Turing machines if they were allowed to operate for an infinite time, but Turing ruled this out with obvious good reason.

Etesi and Nemeti[16] extend this approach by using a pair of computers with one (A) orbiting and another (B) falling towards the event horizon of a Kerr black hole. As the computer (B) approaches the event horizon its time is slowed down relative to the passage of time for (A). The closer it gets to the event horizon the slower its passage of time gets relative to that of (A). They propose that computer (A) be made to work through the infinitely many instances of some recursively enumerable set. An instance they cite is running through the infinitely many theorems of ZFC set theory to see if any are false. If among the infinite set one is found to be false, a light signal is sent to (B) indicating this. Because of the slowdown of time for (B), things can be so arranged as to ensure that the light signal arrives before the event horizon is crossed.

Esti and Nemeti show remarkable ingenuity in working out the details of this scheme. They have a plausible response to the most obvious objection: that the light signal from (A) would be blue shifted to such an extent that (B) could not detect it. They suggest that (A) computes the degree of blue shift that the signal would experience and selects a suitably long wavelength and modulation system to compensate. This is not, however, an adequate response. There remain two serious objections:

(1) Computer (B) is assumed, like any other TM, to operate with clock ticks. The clock cycle is the smallest time within which the machine can respond and carry out any action. There will, within (B)'s frame of reference, be a finite number of clock ticks before the event horizon is crossed. Consider the last clock tick before the horizon is crossed, i.e. the clock cycle that is in progress as the horizon is crossed. Prior to the start of this cycle, machine (A) will have only searched a finite number of the theorems of ZFC set theory. During the final clock cycle of (B) the entire infinite residual of the set of theorems are checked by (A). But any message sent from (A) whilst processing the infinite residual must occupy less than a clock cycle from (B)'s perspective. As such it will be too brief to register at (B).

    Any signal that (B) can respond to will correspond to (A) only having searched a finite part of the infinite set of theorems.

(2) If we consider things from the standpoint of (A), what we are demanding is that it continues to operate reliably, searching through theorems, not just for millions years, but for an *infinite* number of years. The assumptions of infinite reliability and an infinite energy source to keep (A) operating, are clearly impossible.

3 Newtonian computing

Newtonian mechanics is a particularly beguiling area for those exploring the possibilities of infinitary computation. What we call Newtonian mechanics is both that form of abstract maths originating in the calculus of Newton, and a set of laws that allow us to use this mathematical apparatus to make predictive models of reality. The apparatus of the calculus of Newton, with its fluxions and infinitesimals had, from an early stage been controversial [20], a controversy which, in some countries, continued until the late 20th century [21]. It has been known since the early 20th century that, whilst Newtonian mechanics makes excellent predictions of a wide range of physical systems, at extremes of velocity, density and very small scales its success falters. Thus when a paper suggests that Newtonian mechanics allows certain forms of infinitary calculation, what does this mean?

(1) Does it mean that one can logically deduce that certain sets of differential equations will produce infinities in their solutions?
(2) Or does it mean that certain real physical attributes can take on infinite values?
(3) Can we set up physical conditions that will allow particles to undergo infinite accelerations or attain infinite velocities?

In the light of relativity and quantum theory we have to answer the last two questions in the negative whilst asserting 1.

Smith [18] gives as an example of uncomputability in Newtonian physics certain N-body problems involving point masses interacting under gravity. Because these can approach arbitrarily close to one another, at which point their mutual gravitational attraction becomes arbitrarily high, he suggests that one could so configure a set of initial conditions that the particles would move through an infinite number of configurations in a finite time interval. He argues that no Turing machine could simulate this motion in a finite time. This could be interpreted either:

(1) as a limitation on the ability of computers to simulate the world;
(2) or as a means by which, with suitable encoding, a physical system could be used to determine algorithmically uncomputable questions.

But the very properties that would allow this infinitary process - infinite velocities, point masses with position and no magnitude, are the ones where Newtonian mechanics breaks down and has to be replaced with relativistic or quantum mechanics. Smith goes on to show that once relativistic constraints on either velocity or density are introduced, the equations of motion for the system no longer produce infinities, and become algorithmically soluble. It is almost as if there is in the real world a sort of censorship principle that excludes physical laws which could yield infinities.

The infinities in the initial differential equations thus emerge as a consequence of the axiomatic structure of the calculus rather than a property of the real world.

Beggs and Tucker [14] also explore of the extent to which Newtonian mechanics would allow the hypothetical construction of infinitely parallel computing engines. They do not claim that such machines could actually exist, but ask: what sort of mechanics would allow such machines to exist?

On examination though, the mechanics required seem to owe as much to J.K. Rowling as to I. Newton. Beggs and Tucker derive the ability to perform hypercomputation from an infinite plane of conventional computers, which purport to use tricks of Newtonian mechanics to allow infinitely fast transmission of information.

They use two tricks, on the one hand they propose to synchronise the clocks of the machines by using infinitely long, rigid rods. The rod is threaded through all the machines and a push on the rod starts all the computers synchronously. They concede that for this to happen the rod must not only be perfectly rigid, but it must either be massless or have a density which exponentially tends to zero as one moves away from the starting point. This is necessary if the rod is to be moved by applying a finite force. It is not clear in what sense such rods can be said to be Newtonian. One might just as well postulate that you have access to a magic wand that will synchronise an infinite number of computer clocks [2].

They propose that the infinite collection of computers will be able to return results to an initiating processor using an ability to fire cannonballs up and down along parabolic trajectories at arbitrarily high velocities. The arrival of such a cannonball transmits a binary truth value. They further propose that in a finite time interval an infinite number of cannonballs can be projected, in such a way that at any given instant only one is in flight..

Their argument is that given a projectile of mass m we can project it at arbitrarily high speed if we use enough energy. Given a distance b that the cannonball has to travel, we can make the time of flight arbitrarily small by selecting a sufficiently high velocity of travel. The proposal is that one use cannons and an arbitrarily large supply of gunpowder to achieve this. This part of the argument is supposed to accord with Newtonian mechanics, but actually contradicts it on several points.

(1) The immediate problem is that whilst there is according to Newton no limit to the ultimate velocity that a particle subjected to uniform acceleration can reach, this velocity is not reached instantaneously. Suppose we use perfect cannons, ones in which the accelerating force f due to the combustion of gunpowder remains constant along the length of the barrel [3]. A cannonball spends a period in the cannon $\frac{v}{\frac{f}{m}}$ being accelerated that is proportional to the velocity ultimately attained. Thus whilst the flight time $\frac{b}{v}$ tends to zero as velocity increases, total travel time $\frac{b}{v} + \frac{v}{\frac{f}{m}}$ has a finite minimum.

(2) There is a further problem with assuming that cannons can attain an arbitrary velocity. As one increases the charge in a gun an increasing amount of the work done by the powder consists in accelerating the powder itself down the barrel. The limiting velocity achievable is that of the exit velocity of the gas of a blank charge. This, in turn, is limited by the speed of sound in the driving gas. For a highly energetic hydrogen/oxygen explosion this sets a limit of about 2100 m/s. Techniques such as hybrid explosive and light gas guns can produce a limited improvement in velocity[23], but certainly not an arbitrary speed. Rail guns [24] can achieve higher velocities using electromagnetic acceleration. However, since these depend

---

[2] In an analogous context when Turing had recourse to 'Oracles' he deliberately used magical language to indicate that this recourse was make-believe.

[3] Achieving this was the constant, but never achieved goal of 19th century ballistic engineering, much experimentation with powder grain size went into the quest, see for example [22].

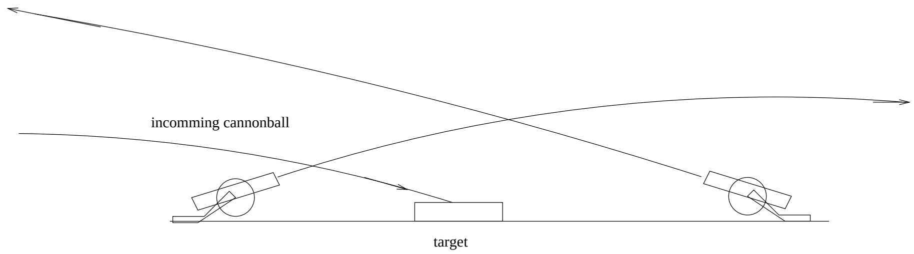

Fig. 2. Cannonballs and their clearances.

on the laws of electromagnetism, which are themselves only valid within a relativistic framework, they are impermissible in Beggs and Tucker's chosen model of mechanics.

(3) Begg and Tucker further assume parabolic trajectories to ensure that the cannonballs fly clear of intervening obstacles like other cannons prior to hitting their targets, see fig. 2. The need to fly clear of other cannons will thus set a minimum angle of fire for the cannons. For a given gravitational acceleration g, and horizontal velocity v, the minimum firing angle θ then sets a finite limit to the time of flight as $\frac{2v\tan\theta}{g}$.

Even on its own terms, leaving aside the question of whether Newtonian physics is realistic, the proposed hypercomputer of Beggs and Tucker is inconsistent with Newtonian mechanics.

4 Analogue hypercomputation

Another theme proposed by advocates of super-Turing computation is the use of analogue computation over real numbers. For a review see [25]. Copeland sees these as being possible if one has available a continuously variable physical quantity which he calls 'charge'. Since he himself recognises that electric charge is discrete, it is not clear how this notional charge is ever to be physically realised. Copeland believes that all the major questions concerning the limits of analogue computation in the real world are open, that nobody knows what might yet be achievable because there has been no systematic investigation, and that such exploration of these topics as did occur was restricted to machines of the differential analyser type.

It is certainly true that the theory of analogue computation is much less developed than that of its digital counterpart, largely for the very pragmatic reason that digital technology came to dominate the entire computing arena. It is undeniable that there is still work to be done in this area as witnessed, for example, by [26]. However, Copeland's latter claim is surely mistaken. This claim is surely mistaken. Differential analysers were a form of mechanical analogue computers, deriving ultimately from Kelvin's well known Harmonic

Analyser[27]. Quite apart from this tradition, there was up to the end of the 1960s an active field of analogue electronic computing. This approach ultimately lost out to digital computers because of the twin problems of accuracy and programmability.

Analogue machines were significantly less accurate than digital ones. Wass[28] stated that

> Errors as small as 0.1% are hard to achieve; errors of 1% are not unusual, and they are sometimes as large as 5%-10%. This is in striking contrast to digital machines, (page 3)

This meant that analogue computers had to be applied in areas where high accuracy was not as important as speed of operation. They were used in aerodynamic control applications such as missile guidance where their errors could be tolerated. In these applications the basic aerodynamic coefficients of the equations being solved were known to only about 10% accuracy, whereas the real-time response needed could not be attained by then existing digital machines.

Analogue machines of this period were programmed by plug boards. Units to carry out operations of adding, subtracting, multiplying, differentiating etc were interconnected by removable wires in the same way as in a manual telephone exchange. Output was typically in the form of real time CRT traces of the graph of the function being computed. Storage of analogue quantities remained a constant problem. MacKay[29] describes the use of modified Williams[30] tubes as analogue stores. Bergman[31] found that over a process of 8 read/write cycles with such the cumulative error amounted to 10%.

Sources of error in such machines were

(1) Systematic scaling errors due to uncertainty in the parameters of the resistors and other passive components being used.
(2) Distortions due to non-linearities in the amplifiers used.
(3) Random fluctuations due to environmental factors. These include thermal noise in the amplifiers, and in the case of Williams tubes the effects of stray magnetic fields on the trajectories of the electrons.

These problems are representative of the difficulties that plague any attempt to carry out accurate analogue computation.

5 Analogue computing with real numbers

In principle an analogue device can be thought of as computing with real numbers. These real numbers can be divided into two classes, the parameters supplied as inputs to the calculation and the variables that take on time varying values as the computation proceeds. Whilst in a digital machine the parameters and the variables are uniformly represented by strings of bits in some digital store, in an analogue machine they are typically of different types. The variables in an electronic analogue computer for example are instantaneous voltages, whilst the parameters are provided by the physical setting of resistors, capacitors etc. These components are subject to manufacturing errors and the provision of higher specification components becomes exponentially expensive.

> That the machine is digital however has a more subtle significance. It means firstly that numbers can be represented by strings of digits that can be as long as one wishes. One can therefore work to any desired degree of accuracy. This accuracy is not obtained by more careful machining of parts, control of temperature variations, and such means, but by a slight increase in the amount of equipment in the machine. To double the number of significant figures, would involve increasing the amount of the equipment by a factor definitely less than two, and would also have some effect in increasing the time taken over each job. This is in sharp contrast with analogue machines, and continuous variable machines such as the differential analyser, where each additional decimal digit required necessitates a complete redesign of the machine, and an increase in the cost by as much as a factor of 10. [32]

The parameters and variables are, in the mathematical abstraction, transformed by operators representing addition/multiplication etc. In an actual analogue computer these operators have to be implemented by some apparatus whose effect is analogous to that of the mathematical operator. The analogy is never exact. Multipliers turn out to be only approximately linear, adders show some losses etc. All of these mean that even were the variables perfect representations of the abstract concept of real numbers, the entire apparatus would only perform to bounded accuracy. But no physically measurable attribute of an apparatus will be perfect representation of a real number.

Voltage for example, is subject to thermal and ultimately quantum noise. We can never set up an apparatus that is completely isolated from the environment. The stray magnetic fields that troubled Bergman will never totally vanish. It may be objected that what we call digital computers are built out of transistors working with continuously varying currents. Since digital machines seem to stand on analogue foundations, what then privileges the digital over the analogue?

The analogue character of, for instance, voltage is itself a simplification. The charge on the gate of an FET is, at a deeper level, derived from an integral number of electrons. The arrival of these electrons on the gate will be subject shot noise. The noise will follow a Poisson distribution whose standard deviation $\approx \sqrt{n}$, with n the mean number of electrons on the gate. It is clear that by raising n, making the device larger, we control the signal to noise level. For large enough transistors this noise can be reduced to such an extent that the probability of switching errors becomes negligible during the normal operation of the computer. It is only if we make devices too small that we have to bear the cost of lower reliability.

Suppose we have a device, either electronic or photonic, that measures in the range 0..1 and that we treat any value above 0.6 as a boolean TRUE and any value below 0.4 as a boolean FALSE and say that in between the results are undefined. Similar coding schemes in terms of voltage are used by all logic families. For simplicity we will assume our measurements are in the range 0 volt to 1 volt.

Now suppose that our switching device is designed to be fed with a mean of 100 quanta when we input a TRUE to it. Following a Poisson distribution we have $\sigma = 10$, so we need to know the probability that the reading will be indeterminate, below 0.6 volt, or how likely is it that only 60 quanta will arrive given shot noise; i.e. a deviation of 4 σ from the mean. Using tabulations of the normal distribution we find that this probability is 0.0000317.

Consider a computer with a million gates each using 100 electrons. Then 31 of the gates would yield indeterminate results each clock cycle. This is unacceptable.

Assuming the million gates, and a 1 Ghz clock and that we will tolerate only one indeterminate calculation a day we want to push this down to a failure probability per gate per cycle of about 10^{-19} or 9σ. This implies $\sigma =$ 0.044 volts which can be achieved when our capacitor is sufficiently large that about 500 electrons will generate a swing of 1.0 volt. The figures are merely illustrative, since they depend on the thresholds of the logic family, but they illustrate the way reliability and number of quanta used to represent TRUE and FALSE trade off against one another.

6 Quantum limits to real number representations

However, all other considerations aside, the idea of being able to represent real numbers physically to any desired degree of accuracy turns out to be in direct contradiction with quantum mechanics. To understand the significance of this statement, it is important to recognise that quantum theory and its extension, quantum field theory, obsolete and replace classical physics, and that this has been so comprehensively demonstrated empirically that it is beyond reasonable doubt. In its fundamental (micro-) nature, the world is not classical, and classical models can never be more than approximations applicable in a limited domain. Recent developments in quantum theory suggest that the classical appearance of the world is, in fact, an artifact of decoherence, whereby the state vector of a quantum system becomes entangled with the environment [33] and that consequently quantum mechanics can indeed be seen as a universal theory governing macroscopic as well as microscopic phenomena. In what follows, we use only the basic axioms of standard quantum theory [34], in particular those governing the deterministic time-evolution of an undisturbed quantum system according to Schrodinger's equation and the discontinuous change of state vector experienced when a system is measured. From these principles it is easy to demonstrate that there are fundamental limits on the.accuracy with which a physical system can approximate real numbers.

Suppose we wish to build an analogue memory based on setting, and later retrieving, some property of a material body, A. To be suitable for analogue representation of real values to arbitrary accuracy, any such property must be continuously variable and so, according to quantum theory, can be represented as a Hermitian operator with a continuous spectrum of real eigenvalues. The most obvious approach to storing a real number in analogue fashion would be to use a positional coordinate of A, with some appropriate degree of precision, as the measurable property. Since only one coordinate is required, in what follows we will assume that A moves in only one dimension and that the value stored is given by its x coordinate, an eigenvalue of the x-position operator $\mathbf{X}$.

It is clear that any such system will have limits imposed by quantum mechanics: the aim here is to establish just how those limits would constrain any conceivable technology. As a first approximation, assume that A is a free body and is not therefore placed in a potential field. A natural scheme might store a real value, say, x_V, at time t_0, by placing A at a point, at distance $x = x_V \pm \Delta x$ from some origin. Δx is the acceptable limitation on the precision of the analogue memory, determined by the physical length of the device and the number of values it is required to distinguish. If 10^R real values ($R \times log_2 10$ bits) are allowed and the maximum value is L (the length of the device), then

$$\Delta x = \frac{L}{2 \times 10^R} \tag{1}$$

We will denote the interval $[x_V - \Delta x, x_V + \Delta x]$ by I_V.

In quantum mechanical terms, just prior to t_0, A is described by a state vector $|\psi_0\rangle$ (Dirac's formalism). Placing A at the chosen point involves "collapsing" $|\psi_0\rangle$ into a new state confined to a positional subspace spanned by the eigenkets $|x\rangle$ with $x \in I_0 = [x_V - \Delta x_0, x_V + \Delta x_0]$. Δx_0 represents the accuracy of the measuring device and it is essential that $\Delta x_0 < \Delta x$ so that $I_0 \subset I_V$ Define $K > 1$, by $K = \frac{\Delta x}{\Delta x_0}$.

This process of "state preparation" is entirely equivalent to performing a measurement on A which leaves it somewhere in the required subspace. Unfortunately, any actual measurement has a non-zero probability of failing to yield a result in I_0. In fact, the probability of success is given by:

$$P\left(x_V - \Delta x_0 < x < x_V + \Delta x_0\right) = \int\limits_{x_V - \Delta x_0}^{x_V + \Delta x_0} \left|\langle x|\psi_0\rangle\right| dx^2 \tag{2}$$

It is reasonably easy to circumvent this problem however. Since the store operation, being a measurement, returns a positional value, it is easy to tell at once if it has failed (if the value lies outside I_0) and we can assume that any number of further attempts can be made until success is achieved. For the sake of simplicity suppose the store operation succeeds on the first attempt at time, t_0, whereupon the new state vector of A is given by:

$$|\psi_s\rangle = \frac{1}{N} \int\limits_{x_V - \Delta x_0}^{x_V + \Delta x_0} |x\rangle \langle x|\psi_V\rangle \, dx \tag{3}$$

where N is a normalisation constant. In wave mechanical terms, this describes a wave packet confined to the region I_0. From the postulates of quantum mechanics, at a time immediately following t_0, a second measurement on A will also retrieve a value within I_0; however, if left undisturbed, $|\psi_s\rangle$ will evolve deterministically, according to the Schrdinger equation and a later measurement will have no such guarantee. The Schrodinger equation can be solved analytically for certain shapes of wave packet, such as the Gaussian but a more general argument is presented below, applicable to a packet of any form. The conclusions are universal and the argument can be extended easily to a packet in a locally constant potential field (e.g. generated by neighbouring atoms). The key point is that, as soon as the wave packet is formed, it begins to spread (dispersion) outside I_0. So long as it remains inside the interval I_V a retrieve operation (measurement) will yield a result in I_V, but once components outside I_V develop, the probability of a read error becomes non-zero and then grows rapidly. Let Δt be the maximum time interval after t_0 during which a measurement is still safe. The analogue memory must be refreshed by performing a new measurement before or an erroneous result may be generated. Note that any real measurement on an interval of length x_V will take at least $\frac{2x_v}{c}$, where c is the speed of light in vacuum, since a light beam must travel to an from A to perform the measurement. It follows that if $\Delta t < \frac{2x_v}{c}$ then the memory will not be feasible.

Since momentum and position are conjugate observables, their x-dimensional operators $\mathbf{X}$ and $\mathbf{P}_x$ obey the commutation relation

$$[\mathbf{X}, \mathbf{P}_x] = i\hbar \tag{4}$$

where $\hbar$ is Planck's constant divided by 2π. From this it follows that the uncertainties (root mean square deviations over a quantum ensemble of identical systems) in the initial (timet_0) values of x and the p_x satisfy the so-called 'Uncertainty Relation'

$$\Delta p_x.\Delta x_0 \geq \frac{\hbar}{2} \tag{5}$$

Since the mass of A written m_A, after the 'store' measurement at t_0, A is moving with an expected velocity of $\frac{\Delta p_x}{m_A}$ away from the prepared position. Naively, therefore, one might expect that at time Δt, A will have traveled a distance of $\frac{\Delta p_x.\Delta t}{m_A}$ away from its prepared position. This however is a quasi-classical argument and underestimates the problem, as one might suspect from noting that the Uncertainty Relation is an inequality and $\frac{\hbar}{2}$ a lower bound. The actual positional uncertainty, Δx, after the critical time Δt (at which the spreading wave packet exceeds its safe bounds), can be obtained via an application of Ehrenfest's Theorem from which it can be concluded ([35] p342) that:

$$\Delta x^2 = \left(\frac{\Delta p_x.\Delta t}{m_A}\right)^2 + (\Delta x_0)^2 \tag{6}$$

In order to determine the theoretical limitations of this system we can now place an upper bound on Δt. Since

(1) $\Delta x = K\Delta x_0$,
(2) $2 \times 10^R \Delta x = L$ and ,
(3) $\Delta p_x \geq \frac{\hbar}{2\Delta x_0}$,

it follows that the maximum value Δt after which the memory becomes unsafe is given by:

$$\Delta t \leq \frac{\sqrt{K^2-1}}{K^2} \times \frac{L^2}{4 \times 10^{2R}} \times \frac{2m_A}{\hbar} \tag{7}$$

It is easy to verify that

$$\max\left(\frac{\sqrt{K^2-1}}{K^2}\right) = \frac{1}{2} \tag{8}$$

Approximating $\hbar \simeq 10^{-34}$ gives,

$$\Delta t \leq 0.25 \times 10^{34-2R} \times L^2 m_A \tag{9}$$

For viability, $\Delta t \geq \frac{2x_V}{c}$ so, since $c \approx 3 \times 10^8 (ms^{-1})$:

$$m_A > \frac{8 \times 10^{2R-42}}{3L} \times \frac{x_V}{L} \tag{10}$$

This depends on the value being stored. However, the memory must work for all values of x_V so we can set $x_V = L$. We now have a final result:

$$m_A \times L > 2.6 \times 10^{2R-42} \tag{11}$$

or alternatively

$$m_A \times \Delta x > 1.3 \times 10^{R-42} \tag{12}$$

The limitation is is applicable to any scheme which relies on the physical position of an object to store a real number. To increase the range of a positional analogue representation by 1 bit of precision requires the mass$\times$length product to increase by approximately a factor of 4. It is very easy to see that a system of this type cannot scale unless allowed to grow arbitrarily in physical size, with consequent implications not only for the feasibility of construction but also for the speed at which stored data might be retrieved.

In contrast to a digital machine where the resources used to perform higher accuracy computation grow linearly with the number of bits of accuracy, the resources needed by the analogue machine grow exponentially with the accuracy of the calculation.

The following table gives approximate values for the parameters required for one-dimensional analogue positional memories at several values of precision (in bits), where possible for L=10m. The 32-bit example requires masses of the order of a few atoms of uranium placed to an accuracy of 2.5nm, technically not an outrageous scenario.

Precision ($R \log_2 10$)	Length	Approx Mass	$2\Delta x$
32	10m	5×10^{-24}kg	2.3×10^{-9}m
64	10m	10^{-4}kg	5.4×10^{-19}m
128	5km	6×10^{31}kg	1.4×10^{-35}m

However, above 32-bit precision the systems become increasingly implausible even with exotic technologies. For the 128 bit example, L is chosen to ensure that Δx exceeds the Planck Length, L_p (1.6×10^{-35}m), which is the minimum precision possible in any physical measurement of position. However even accommodating this fundamental constraint,m_A is of the order of 50 solar masses and must occupy a space at least equal to its Schwarzschild radius, S_A. As this is given by:

$$S_A = m_A \times 1.48 \times 10^{-27} \tag{13}$$

A would be at least several km across even if it was a black hole. When dealing with objects with sufficient gravitational fields, the Uncertainty Principle has to be modified to take account of gravitational forces and, while the detail is still not fully settled, in most current theories of quantum gravity it is believed[36] to take a form such as:

$$\Delta x \geq \frac{\hbar}{2\Delta p} \times k L_p^2 \frac{\Delta p}{\hbar} \tag{14}$$

where k is a constant. This inequality (the so-called Generalised Uncertainty Principle) is interesting because it predicts that when dealing with massive objects that the Δx associated with a given Δp may significantly higher than in the non-gravitational case. The GUP also confirms that the uncertainty in position can never be less than L_p regardless of how imprecise the momentum knowledge is: this is in agreement with the claim that L_p is the smallest measurable quantum of length. We conclude:

(1) Any physically build-able analogue memory can only approximate the reals and there are very definite limits as to the accuracy achievable.
(2) Analogue storage of reals, will for high precision work, always be outperformed in terms of device economy by digital storage.
(3) Physically build-able analogue computers can not rely upon the availability of exact stored real numbers to outperform digital computers.

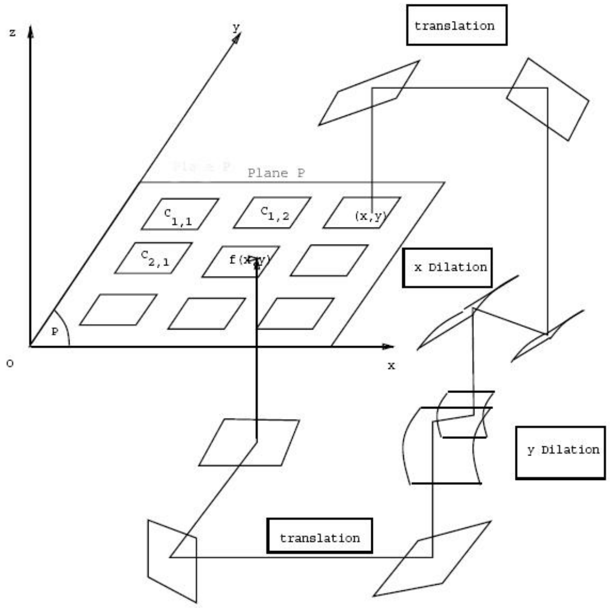

Fig. 3. An analogue computer proposed by Bournez and Cosnard. Reproduced from [17].

7 Bournez and Cosnard's analogue super-Turing computer

An examination of a concrete example of a proposed super-Turing analogue computer[17] illustrates the sorts of errors that would vitiate its operation. Bournez and Cosnard propose to use two real valued variables corresponding to the x, y coordinates of particles, (presumably photons) passing through plane P in Figure 3. The binary expansion of these real valued coordinates could then be used to emulate the left and right parts of a TM tape [37,38]. They argue that the machine could, in addition, be used to simulate a class of two stack automata whose computational powers might exceed those of TMs. The gain in power comes from the ability of their proposed stack automaton to switch on the basis of the entire contents of an unbounded stack, rather than on the basis of what is directly under the TM head. They suggest that if one had available iterated real valued functional systems based on piecewise affine transforms, such analogue automata could be implemented. In the proposed physical embodiment given in Figure 3, multiplication by reals would be implemented by pairs of parabolic mirrors, and translation by arrangements of planar ones.

The authors, in striking contrast to researchers active in the area 50 years earlier, fail to identify the likely sources of error in their calculator. Like any other analogue system it would be subject to parameter, operator and variable errors. The parameters of the system are set by the positions and curvatures of the mirrors. The placement of the mirrors would be subject to manufacturing errors, to distortions due to temperature, mechanical stress etc. The parabolic mirrors would have imperfections in their curvature and in their surfaces. All of these would limit the number of significant digits to which the machine could calculate. But let us, for the moment, ignore these manufacturing errors and concentrate on the inherent uncertainty in the variables.

Because of the wave particle duality any optical system has a diffraction limited circle of confusion. We can say that a certain percentage of the photons arriving from a particular direction will land within this circle. The radius of the circle of confusion is inversely proportional to the aperture of the optical system and directly proportional to the focal length of the apparatus and to the wavelength of the photons used. The angle to the first off-center diffraction peak $\Delta\Theta$ is given by

$$sin(\Delta\Theta) = \frac{\lambda}{A} \tag{15}$$

where A is the aperture and λ the wavelength.

By constraining the position of the photon to be within the aperture, we induce, by the principle of Heisenberg, an uncertainty in its momentum within the plane of the aperture.

To see what this implies, we give some plausible dimensions to the machine. Assume the aperture of the mirrors to be 25mm and the path length of a single pass through the system from the first mirror shown in Figure 3 back to Plane P to be 500mm. Further assume that we use monochromatic light with wavelength $\lambda = 0.5\mu$. This would give us a circle of confusion with a a radius $\Delta_{f(x,y)Mirror} \approx 10\mu$.

If plane P was $\frac{1}{10}$ of a meter across the system would resolve about 5000 distinct points in each direction as possible values for *f(x, y)*. This corresponds to about 12 bits accuracy.

The dispersion $\Delta_{f(x,y)Mirror}$ accounts only for the first pass through the apparatus. Let us look at the parametric uncertainty in x, y to start out with.

One wants to specify x, y to greater accuracy than *f(x,y)* so that $\Delta_{x,y} < \Delta_{f(x,y)}$. Assume we have a source of collimated light whose wavefronts are normal to the axis of the machine. A mask with a circular hole could then constrain the incoming photons to be within a radius $< 10\mu$. Any constraint on the position of the photons is an initial aperture. If this aperture $\leq 10\mu$, its diffraction cone would have a $\Delta\Theta \approx 0.05$ radians. Going through the full optical path the resulting uncertainty in position $\Delta_{f(x,y)Mask(\Delta_{x,y})} \approx 25$mm. We have $\Delta_{f(x,y)Mask(\Delta_{x,y})} >> \Delta_{f(x,y)Mirror}$.

The narrower the hole in the mask the more uncertain will be the result *f(x,y)*. In other words the lower the parametric error in the starting point, the greater the error in the result.

There will be a point at which the errors in the initial position and the errors due to the diffraction from the mirrors balance: when $\Delta_{f(x,y)Mirror} + \Delta_{f(x,y)Mask(\Delta_{x,y})} \leq \Delta_{x,y}$. From simple geometry this will come about when the ratio $\Delta_{x,y}/L \approx \lambda/\Delta_{x,y}$ so

$$\Delta_{x,y} \approx \sqrt{L\lambda} \tag{16}$$

where L is the optical path length of the computation. For the size of machine which we have assumed above, this implies $\Delta_{x,y} = 500\mu$. Its accuracy of representation of the reals is thus less than 8 bits(= - $\log_2(\frac{500\mu}{100mm})$), hardly competitive with existing digital computers.

The evaluation of *f(x,y)* corresponds to a single step of a TM program. If n is the number of TM steps, optical path length is nL. By (16), the optimal initial aperture setting $\Delta_{x,y} \propto \sqrt{n}$. Each fourfold increase in the execution length of the program, will reduce by 1 bit the accuracy to which the machine can be set to compute.

If we want to make an optical machine more accurate we have to make it bigger - the growth in the size of astronomical telescopes bears witness to this. For every bit of accuracy we add, we double the linear dimensions. If M is the mass of the machine, and b its bit accuracy then $M \propto 2^{3b}$.

For a conventional digital VLSI machine, $M \propto b$ and the mass of the arithmetic unit grows as $b \log b$. For any but the least accurate calculations this sort of optical analogue machine will be inferior to conventional machines.

8 Quantum Computers

Deutsch[39] and Feynman[40,41] opened up the recent field of quantum computing research. An important result obtained by Deutsch was that although quantum machines could potentially have higher parallelism than classical ones, they would not extend effective computation beyond the recursive functions.

Kieu [42] recently challenged Deutch's result, with a proposal to solve Diophantine equations by means of a quantum process. The basic idea is to prepare a quantum system with a set of possible states mapped to natural numbers and a Hamiltonian which encodes the Diophantine equation. The system would be allowed to adiabatically evolve to a ground state corresponding to a solution to the equation. This proposal, however, remains controversial. Critical responses include [43–45] and replies to these criticism [46,47]. Points made by critics are:

(1) There is no guarantee that one can distinguish between a system that has settled into a true ground state or a 'decoy' false minimum.
(2) One would need arbitrary high precision in energy measurements.
(3) The evolution time needed to reach the ground state is unknown and in principle Turing uncomputable. One thus reproduces the Halting Problem in a different form.

9 Conclusions

For Turing, computers were not mere mathematical abstractions but concrete possibilities, which he pursued in his designs for the Bombe at Bletchley and ACE at the National Physical Laboratory[32,48]. Computing Science has undergone explosive progress in the period since Turing's original paper, synthesising abstract computational theory with research into possible physical implementations of digital computers. An enormous volume of research has produced successively more effective generations of such machines. The development of smaller and faster computers has been a constant struggle towards the limits of what is physically possible.

While one cannot say in advance how small or how fast we will eventually be able to build computers, we do know that any effective computer we build must be bound by the laws of physics, for example the fundamental thermodynamic limitations of computation explored by Landauer[19,49].

In this paper we have surveyed appeals to deployments of actualised infinities in time and space resources, and of continua in representations, to transcend the computational limits established by the proof of the undecidability of the entscheidungsproblem 60 years ago. In particular, we have discussed attempts to use:

- infinitesimal communication times;
- arbitrary precision real number representations;
- Newtonian computing;
- quantum computing.

In our assessment, appeals to infinity in new models of computation run up against fundamental physical limits.

Acknowledgments

This paper develops and extends arguments first presented in [7] and [8].

References

[1] A. M. Turing, On Computable Numbers, With an Application to the Entscheidungsproblem, Proceedings of the London Mathematical Society 42 (1937) 230–65.

[2] A. Church, An unsolvable problem of elementary number theory, American Journal of Mathematics 58 (1936) 345–363.

[3] A. M. Turing, Systems of logic based on the ordinals, Proceedings of the London Mathematical Society 45 (1939) 161–228.

[4] S. C. Kleene, General recursive functions of natural numbers, American Journal of Mathematics 57 (1935) 727–742.

[5] E. L. Post, Finite Combinatory Processes. Formulation 1, Journal of Symbolic Logic 1 (1936) 103–105.

[6] J. Copeland, Accelerated Turing Machines, Minds and Machines 12 (2002) 281–301.

[7] G. Michaelson, P. Cockshott, Constraints on Hypercomputation, in: A. Beckmann, U. Berger, B. Lowe, J. V. Tucker (Eds.), Logical Approaches to Computational Barriers: Second Conference on Computability in Europe, CiE 2006, Swansea, UK, no. 3988 in LNCS, Springer, 2006, pp. 378–387.

[8] P. Cockshott, G. Michaelson, Are there New Models of Computation: a Reply to Wegner and Eberbach, Computer Journal To appear.

[9] P. Wegner, E. Eberbach, New Models of Computation, Computer Journal 47 (2004) 4–9.

[10] Aristotle, The Politics, Hutchinson, 1988.

[11] A. Jeffrey, Mathematics for Scientists and Engineers, Nelson, 1969.

[12] L. Althusser, É. Balibar, Reading Capital, New Left Books, 1970.

[13] J. Lassegue, Turing, entre formel et forme ; remarques sur la convergence des perspectives morphologiques, Intellectica 2002/2 (35) (2002) 185–198.

[14] E. Beggs, J. Tucker, Embedding infinitely parallel computation in Newtonian kinematics, Applied mathematics and computation 178 (1) (2006) 25–43.

[15] J. D. Hamkins, Infinite Time Turing Machines, Minds and Machines 12 (4) (2002) 521–539.

[16] G. Etesi, I. Németi, Non-Turing Computations Via Malament–Hogarth Space-Times, International Journal of Theoretical Physics 41 (2) (2002) 341–370.

[17] O. Bournez, M. Cosnard, On the computational power and super-Turing capabilities of dynamical systems, Tech. Rep. 95-30, Ecole Normal Supérior de Lyons (1995).

[18] W. SMITH, Church's thesis meets the N-body problem, Applied mathematics and computation 178 (1) (2006) 154–183.

[19] R. Landauer, Irreversibility and heat generation in the computing process, IBM Journal of Research and Development 5 (1961) 183–91.

[20] J. Wisdom, Berkeley's Criticism of the Infinitesimal, The British Journal for the Philosophy of Science 4 (13) (1953) 22–25.

[21] J. Dauben, Marx, Mao and Mathematics: The Politics of Infinitesimals, Documenta Mathematica 3 (1998) 799–809.

[22] T. SCHENCK, MANUFACTURE OF GUNPOWDER, US Patent 273,209 (Feb. 27 1883).

[23] W. Crozier, W. Hume, High-Velocity, Light-Gas Gun, Journal of Applied Physics 28 (8) (1957) 892–894.

[24] S. Rashleigh, R. Marshall, Electromagnetic acceleration of macroparticles to high velocities, Journal of Applied Physics 49 (4) (1978) 2540–2542.

[25] J. Copeland, R. Sylvan, Beyond the universal Turing machine, Australasian Journal of Philosophy 77 (1) (1999) 46..66.

[26] J. V. Tucker, J. I. Zucker, Computability of analogue networks, Theoretical Computer Science 371 (2007) 115–146.

[27] W. Kelvin, Harmonic analyzer., Proceedings of the Royal Society of London, 27 (1878) 371–373.

[28] C. A. A. Wass, Introduction to Electronic Analogue Computers, Pergamon, 1955.

[29] D. MacKay, M. Fisher, Analogue Computing at Ultra High Speed, Chapman and Hall, London, 1962.

[30] F. C. Williams, A Cathode Ray Tube Digit Store, Proceedings of the Royal Society of London 195A (1948) 279–284.

[31] G. D. Bergman, A New Electronic Analogue Storage Device, in: Proceedings of the International Analogy Computation Meeting, Presses Academiques Europeennes, Brussels, 1955, pp. 92–95.

[32] A. M. Turing, Lecture on the Automatic Computing Engine, 1947 , in: B. J. Copeland (Ed.), The Essential Turing, OUP, 2004.

[33] R. Omnes, The interpretation of quantum mechanics, Princeton University Press, NJ, 1994.

[34] B. d'Espagnat, Conceptual Foundations of Quantum Mechanics, Addison-Wesley, 1976.

[35] C. Cohen-Tannoudji, B. Diu, F. Laloë, S. Hemley, Quantum mechanics.: Vol. 1, Wiley Interscience;, 1977.

[36] F. Scardigli, Generalised Uncertainty Principle in Quantum Gravity from Micro-Black Hole Gedanken Experiment, Physics Letters B 452 (1999) 39–44.

[37] P. Koiran, C. Moore, Closed-form analytic maps in one and two dimensions can simulate Turing machines, Theoretical Computer Science 210 (1) (1999) 217–223.

[38] L. A. Bunimovich, M. A. Khlabystova, Lorentz lattice gases and many-dimensional Turing machines, Collision-Based Computing 443–468.

[39] D. Deutsch, Quantum theory, the Church-Turing principle and the universal quantum computer, Proceedings of the Royal Society of London A 400 (1985) 97–117.

[40] R. P. Feynman, There's plenty of room at the bottom, in: A. Hey (Ed.), Feynman and Computing, Westview Press, 2002.

[41] R. P. Feynman, Simulating physics with computers, in: A. Hey (Ed.), Feynman and computation: exploring the limits of computers, Perseus Books, Cambridge, MA, USA, 1999, pp. 133–153.

[42] T. D. Kieu, Quantum algorithm for Hilbert's tenth problem, International Journal of Theoretical Physics 42 (2003) 1461 – 1478.

[43] W. D. Smith, Three counterexamples refuting Kieu's plan for "quantum adiabatic hypercomputation" and some uncomputable quantum mechanical tasks, J.Applied Mathematics and Computation 187 (1) (2006) 184–193.

[44] A. Hodges, Can quantum computing solve classically unsolvable problems?, Tech. Rep. quant-ph/0512248, arXiv.org (Dec 2005).

[45] M. Davis, The Church-Turing Thesis: Consensus and Opposition, in: A. Beckmann, U. Berger, B. Lowe, J. V. Tucker (Eds.), Logical Approaches to Computational Barriers: Second Conference on Computability in Europe, CiE 2006, Swansea, UK, no. 3988 in LNCS, Springer, 2006, pp. 125–132.

[46] T. D. Kieu, Reply to Andrew Hodges, Tech. Rep. quant-ph/0602214, arXiv.org (Feb 2006).

[47] T. D. Kieu, On the identification of the ground state based on occupation probabilities:, Tech. Rep. quant-ph/0602145, arXiv.org (Feb 2006).

[48] A. Hodges, Alan Turing the enigma, Touchstone, 1983.

[49] R. Landauer, Information is Physical, Physics and Computation, 1992. PhysComp'92., Workshop on (1992) 1–4.