EJTP **13, No. 36** (2016) 169–182 *Electronic Journal of Theoretical Physics*

Advances in Three Hypercomputation Models

Mario Antoine Aoun*

Department of the Phd Program in Cognitive Informatics, Faculty of Science, Université du Québec à Montreal (UQAM), QC, Canada

Received 1 March 2016, Accepted 20 September 2016, Published 10 November 2016

Abstract: In this review article we discuss three Hypercomputing models: Accelerated Turing Machine, Relativistic Computer and Quantum Computer based on three new discoveries: Superluminal particles, slowly rotating black holes and adiabatic quantum computation, respectively.
© Electronic Journal of Theoretical Physics. All rights reserved.

Keywords: Hypercomputing Models; Turing Machine; Quantum Computer; Relativistic Computer; Superluminal Particles; Slowly Rotating Black Holes; Adiabatic Quantum Computation
PACS (2010): 03.67.Lx; 03.67.Ac; 03.67.Bg; 03.67.Mn

1 Introduction

We discuss three examples of Hypercomputation models and their advancements. The first example is based on a new theory that advocates the use of superluminal particles in order to bypass the energy constraint of a physically realizable Accelerated Turing Machine (ATM) [1, 2]. The second example refers to contemporary developments in relativity theory that are based on recent astronomical observations and discovery of the existence of huge slowly rotating black holes in our universe, which in return can provide a suitable space-time structure to operate a relativistic computer [3, 4 and 5]. The third example is based on latest advancements and research in quantum computing modeling (e.g. The Adiabatic Quantum Computer and Quantum Morphogenetic Computing) [6, 7, 8, 9, 10, 11, 12 and 13].

Copeland defines the terms Hypercomputation and Hypercomputer as it follows: "Hypecomputation is the computation of functions or numbers that [...] cannot be computed with paper and pencil in a finite number of steps by a human clerk

* Email: aoun.mario@courrier.uqam.ca, mario@live.ca

working effectively…A hypercomputer is any information-processing machine, notional or real, that is able to achieve more than the traditional human clerk working by rote…Hypercomputers compute functions or numbers, or more generally solve problems or carry out tasks, that lie beyond the reach of the Universal Turing Machine" [1].

Up until the year 2010, Hypercomputation did not have a solid grounding in the realm of engineering, however this fact has not just changed but even more, ameliorated with the development of the 'Infinity Machine' [7] (Technically, Adiabatic Quantum Computer; which its commercial version was first released in 2011, by D-Wave company based in Vancouver, Canada) that can find solutions, in a matter of days, for optimization problems that require years to solve if executed on standard; even parallel processing, computer models [7, 8 and 10]. Besides, in Science, every theory has its exceptions and limitations that bind it to the domain of its application. In the computing discipline, we should look at where limitations can be exploited in order to open the doors for new developments and give rise for possible new theories. For instance, what if the energy consumption to perform an operation on an Accelerated Turing Machine (ATM) does not grow exponentially with the number of operations to be executed? [2]. Thus, by just reflecting on the hypotheses upon which we conclude our results (e.g. a physical realization of an ATM is impossible) and by expanding the frontiers of the domain of applicability (e.g. The Machine upon which we implement our attempt), then finding solutions for unsolvable problems could be possible. It can be thought unreasonable of considering such efforts and conjectural challenges; like rejecting pre-set hypotheses, objecting old theories whilst building new theories in the aim to solve problems that their solutions are already comprehended as unattainable. However, this is not the case in the scientific practice, which its virtue relies on questioning, exploration and discovery. For instance, let's consider the proof of Fermat last Theorem. This theorem took 700 years to be solved. Its solution by Andrew Wiles in 2001 gave rise to new developments in the field of mathematics that make a link between Galois representations, Modular forms and L-functions; which are distinct areas in number theory [14]. We note that the solution of Fermat Last Theorem has ever been deemed to be impossible. The mathematician Kenneth Ribet said that with the proof of Fermat last theorem "the mathematical landscape has changed…You discover that things that seemed completely impossible are more of a reality" [15].

Moreover, a *Super Task* is a task, which requires an infinite number of operations to be performed in a finite amount of time [16]. For instance, imagine a computer model that can execute an infinite number of statements (translated into logical and arithmetical operations) in a finite amount of time. This seems to be impossible. However, one theoretical model; was envisaged in such direction, it is called Accelerated Turing Machine (ATM) [1] and described next (Section 2). Furthermore, in section 3, we will discuss a new insight for physically realizing ATMs using Superluminal particles. Then we will tackle new advancements in Relativistic computing

(Section 4) and Quantum Computing modelling (Section 5) that attempt to execute super tasks. Section 6 concludes the article with Critics by Martin Davis.

2 Accelerated Turing Machines (ATM)

One theoretical model that can achieve a *Super Task* is called Accelerated Turing Machine (ATM) [1]. An ATM executes a single operation in the half amount of time taken to execute the previous operation [1]. This can be explained with the following mathematical formula:

$$ET\ (t) = \frac{ET\,(t-1)}{2} \tag{1}$$

Where ET means Execution Time and t is the time step; which can be defined in any unit of time (e.g. milliseconds, seconds, minutes...)

With initial conditions: $ET(0) = 1$.

In the scenario of executing a super task, the Total Execution Time (TET) of the task; which requires t to go to infinity, is the limit of this sequence at infinity and can be written as follows:

$$\begin{aligned}
TET &= \sum_{i=0}^{t} ET(i) \\
TET &= ET\,(0) + ET\ (1) + ET\,(2) + ET\,(3) + ... \\
TET &= 1 + \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + ... \\
TET &\cong 2
\end{aligned} \tag{2}$$

Since the limit of this sequence converges to 2, then the time steps required to execute a super task; that requires an infinite number of operations, converges to two time units. In this mode of operation, an ATM will execute an infinite number of operations in a finite amount of time. This can be considered as a model of a Hypercomputer in a pure theoretical sense [1]. Next, we will study a possible physical realisation of this theoretical ATM.

3 ATM using Superluminal Particles

In this section, we will first discuss why ATM was considered not to be physically realizable, and then we will discuss how it can be physically realizable by considering the use of the recent theory of Superluminal Particles [2].

According to Takaaki Musha [2], ATM is considered not to be physically realizable "because the energy to perform the computation will be exponentially increased when the computational step is accelerated" [2]. This is based on the energy-time uncertainty relation [17, 18] given next:

$$\Delta H \Delta T \sim \hbar \tag{3}$$

Where, ΔH is "the standard of energy, ΔT is a time interval" [17] and $\hbar$ is the planck constant h divided by 2π.

In order to realize hypercomputation in the means of an ATM, Musha [2] suggests the use of superluminal particles instead of subliminal particles (e.g. Photons) to escape the uncertainty of energy-time relation (equation 3). Musha bases his work on the energy consumption of the reversible computer (e.g. Richard Feynman model), which is a quantum computer. He refers to Lloyd influential paper on the physical limits of computation [19] by showing that the quantum system; based on the study of its energy consumption per computational step as given by Feynman, requires an infinite amount of time to complete infinite steps of computation when the system uses subluminal particles, as photons [2]. Hence, Musha proceeds by studying the uncertainty principle based on superluminal particles. Superluminal means traveling faster that light. By the means of quantum tunneling, where photons tunnels a material or a barrier, it was shown that a photon could travel faster than light when it tunnels a specific barrier or material. This means that the speed of photons can by pass the celerity of light (c) when they are inside a specific barrier or material. In this regard, Musha quotes the studies of [20] and the experiments of [21, 22 and 23] which proves superluminal behavior based on quantum tunneling. Musha reaches a new uncertainty relation of energy-time for superluminal particles, which is:

$$\Delta H \Delta T \sim \frac{\hbar}{\beta.(\beta - 1)} \tag{4}$$

Where, $\beta = \frac{v}{c}$

c is the celerity of light and v is the speed of the particle after it has been measured (Note that the speed of the particle is equal to c before the measurement).

Last, Musha derives the equation of the total computational time required by a quantum system based on a Feynman model that uses superluminal particles when the system performs a number of computational steps. He finds out that the total computational time converges while the number of computational steps that the system executes tends to infinity. This means, "an accelerated Turing machine can be realized by utilizing superluminal particles instead of subliminal particles for the Feynman's model of computation" [2].

4 Relativistic Computers

In the context of hypercomputation and the time required for a computer to fulfill an infinite number of operations in a finite amount of time, Copeland states Hogarth who says "there is no reason why a computer user must remain beside the computer" [1]. This elaborates the idea of relativistic computing where a computer executing a task can travel in spacetime and transmits the answer of its computation to a user. In this regard, for a specific spacetime that is considered to be a relativistic spacetime, "the infinite lifespan of the computing machine can be surveyed by the user in a finite amount of time" [1].

New discoveries in relativity theory, black hole physics and cosmology suggests that the possibility of a relativistic computer is scientifically adequate and can be physically realizable in our existing universe [3]. To study and explain the novel relativistic computer model, we will refer to the work of Andreka et al. [3]; who claim: "In our specific physical universe there seem to exist regions of spacetime supporting potential non-Turing computations" [3]. In comparison to quantum computing that tackles the complexity barriers of the theory of computation, the relativistic computing paradigm tackles and challenges the physical Church Turing thesis itself. The physical Church Turing thesis says that a physical Turing machine can simulate any algorithmic function. The physical Turing machine is a mere physical device, which was accentuated on the worldview of Newtonian physics where the notion of time is considered to be absolute [3]. This is not the case in the context of relativity theory where time is relative to its observers. This means, "Various observers at various points of spacetime in different states of motion might experience time radically differently. Therefore, we might be able to speed up the time of one observer, say C (Cecil, for 'computer'), relatively to the other observer, say P (Peter, for 'programmer'). Thus P may observe C computing very fast" [3]. The experiments of [24, 25] study the spacetime structure of huge slowly rotating black holes. The confirmation of the existence of these black holes is based on recent astronomical observations [4, 5]. Note that such black holes are called slow-Kerr black holes in the literature.

Black holes are regions of spacetime that exhibit a very high gravitational field. The following example illustrates the idea of speeding up computation: Consider an extremely very high tower that exists on earth. Also, consider two atomic clocks, one running on the top of the tower and the other on the bottom. According to the Gravitational Time Dilation (GTD) theorem of relativity theory, the clock on the top will run faster than the clock on the bottom because gravity affects time. As much as gravity is strong, the time is slow. The clock on the bottom will run slower than the clock on the top because gravity makes time run slower. So, P can send his computer C to the top to gain execution time from this speed time effect [3]. Furthermore, if "we want to increase this speed-up effect to the infinity. Therefore, instead of the Earth, we use a huge black hole" [3]. Andreka considers the slow-Kerr black holes (i.e. huge slowly rotating black holes) that have two event horizons of bubble like surfaces. These two event horizons are the inner event horizon and the outer event horizon as illustrated in the following drawing next (Fig. 1).

Respectively, the earth is the slowly rotating balck hole, the programmer P on the bottom of the tower is considered to be on the inner event horizon and the computer C on the outer event horizon. "If we could suspend the lower observer P on the event horizon itself then from the point of view of C, P's clocks would

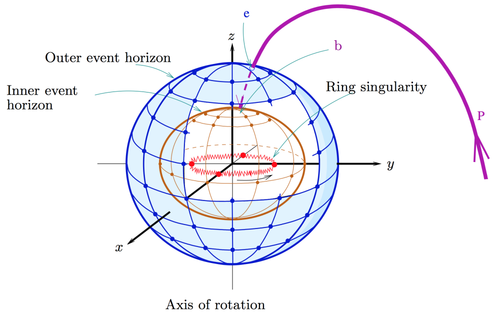

Fig. 1 Depiction of a slowly rotating black hole showing its outer event horizon and its inner event horizon. Because the black hole is rotating, then the inner event horizon exerts a centrifugal force on P in the Newtonian sense, which in return can suspend the motion of P. This is a main characteristic of slowly rotating black holes (i.e. having two event horizons in contrast to Schwarzschild black holes which have only one event horizon that could crush P). Excerpt from [3] with permission from the authors.

freeze, therefore from the point of view of P, C's clocks (and computers!) would run infinitely fast, hence we would have the desired infinite speed-up upon which we could then start our plan for breaking the Turing barrier" [3]. An appropriate geodesic can be chosen for P so it enters the black hole and bypass the inner horizon. Once inside the inner horizon, P can remain in constant distance to C; that is moving on the outer horizon, due to the centrifugal force of the rotating black hole that is exerted on P. Now, any information sent by C to P can "reach P before P meets the inner horizon" [3] as seen in Fig. 2.

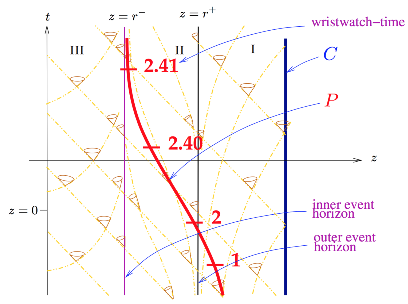

Fig. 2 Showing z as the axis of the rotating black hole and t as time. The world lines of C and P are shown as well as the inner and outer event horizons separated by τ. A photon that is sent by C to P arrives to P before P reaches the inner event horizon. Excerpt from [3] with Permission from the authors.

As seen in Fig.2, "the time measured by P is finite . . . while the time measured by C is infinite" [3].

The approach of considering slowly rotating black holes is interesting and provides advancement in the theory of a relativistic computer. Its major importance is that a slowly rotating black hole has two event horizons, which is different than a simple black hole like a Schwarzschild black hole that have only one event horizon. By considering only one event horizon, we don't have a centrifugal effect, thus P cannot slows down. Therefore, the gravitational force of the black hole will crush

P [3]. Furthermore, we have to note that the advantage of choosing slowly rotating black holes compared to simple black holes is that P will slow down by just using the gravitational force of the inner horizon without the need of brute force like using a rocket to slow down its fall or maintain its altitude [3].

5 Quantum Computing

In this section, we first introduce Quantum Computing as a possible model of Hypercomputation and then discuss its latest advancements in this regard.

According to Kieu [26], "It is argued that computability, and thus the limits of Mathematics, ought to be determined not solely by Mathematics itself but also by physical principles" [26]. In his paper, Kieu studies a quantum algorithm to solve the 10^{th} Hilbert problem of finding the existence of solutions of Diophantine equations. Kieu's argument suggests that the realization of a quantum algorithm could impact the standards of the Church-Turing thesis and will modify its effectiveness in terms of computability. He further argues that the proof of the realization is constrained to physical limitations not logical argument [26]. To give insight on the plausibility of quantum computing that could be advanced from this quantum algorithm he quotes Godel in [27] who says: "... On the other hand, on the basis of what has been proved so far, it remains possible that there may exist (and even be empirically discoverable) a theorem-proving machine which in fact is equivalent to mathematical intuition, but cannot be proved to be so, nor even be proved to yield only correct theorems of finitary number theory." Kieu suggests that quantum computation is that possibility.

Richard Feynman [6] was the first person who questioned the plausibility of building physical computing devices that could operate on quantum rules and follow the principles of quantum mechanics based on the laws of quantum physics rather than classical physics. With Feynman, a contemporary era of the theory of computation is born that resembles the modern era of computing that was born with Alan Turing influential work on the theory of computation in the mid thirties.

Computers nowadays are an elaboration of the Turing machine. The latter process information in the form of bits. This means that a bit (i.e. an entity that takes the value of either 0 or 1) is the single unit of information that can be processed within a computer. Furthermore, an elementary operation of a computer processes a single bit at a time step. In contrast, in a quantum computer, information is encoded in qubits. A qubit is a single unit of information that encapsulates the state of a bit by being 0 or 1; moreover it can take another value that is 0 and 1 superposed! This fact is due to the superposition principle of quantum mechanics, which considers that an electron (i.e. a particle) can be in any state unless it is measured, thus it can be in a superposition of states. So, under quantum rules the single unit of information in a quantum computer can be 1, or 0, or 1 and 0 at the same time. As David Deutsch explains in his book "The Fabric of Reality", a quantum bit is the dual existence of a bit, where each of its states exists in different universes [28]. An illustration of a bit and a qubit is given in Fig. 3 next.

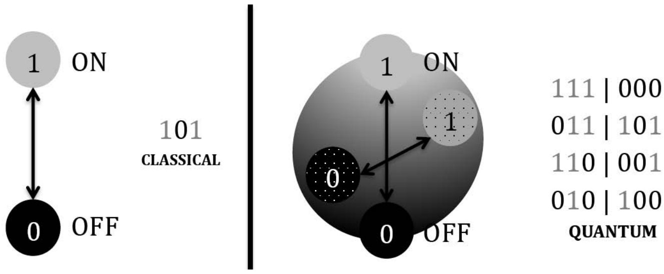

Fig. 3 This shows an illustration of a classical bit vs. a quantum bit and the potential of using quantum bits to perform computation. On the left, data units take the form of a bit that is processed in a classical computer as an entity having a single state; which is either 0 or 1. On the right, data units take the form of a quantum bit (i.e. qubit) that can be 0, 1 or 0 and 1 at the same time due to the principle of quantum superposition. A data unit exhibiting multiple states allows concurrent operations that lead fast data processing, which is achieved using a quantum computer model. Illustration re-sketched from [7].

The latest advancement in quantum computing is a quantum physical computer called the D-wave machine [7]. In the common sense, this quantum computer is called the Infinity Machine [7] and it is now suggested as a significant reference for the engineering community [8]. This device (namely D-Wave two) is the latest of its kind that instruments a physical realization of a quantum-computing machine, which is able to perform tasks (e.g. Solving optimization problems) that defeat the performance of a classical computer. Moreover, it outperforms the super scale of ultimate parallel computation. This quantum computer is being implemented in Vancouver, Canada. In other words, it physically exists in a computer company called D-Wave and it operates in a cooling room on a temperature of -459.6^0 F, inside the firm. It is a device of 10 ft High and its central processing unit is called the niobium chip.

How D-wave quantum computer work?

D-wave quantum computer is able to solve optimization problems and is based on Quantum Tunneling. Quantum tunneling is the phenomenon in which a particle can cross a barrier with a specific probability. Since solving an optimization problem is thought like finding the lowest points in a mountains landscape, then think of the particles as tunneling through the mountains which is in contrast with classical mechanics where a particle could either bounce back of a hill or get crushed inside the mountain. The valleys found by the tunneling particles are considered as candidate solutions. Quantum Annealing and Adiabatic Quantum Computation, described next, are methods that can be applied to the tunneling phenomenon in order to find the best candidate solution (i.e. the lowest valley).

In Quantum Annealing (QA), we try to minimize an Energy function which is represented by a Hamiltonian h. Furthermore, we add to h a local Hamiltonian $h_L(t)$ which starts very large and reduces to 0 after n time steps. In other words, the total Hamiltonian is:

$$H\,(\mathrm{t}) = h + h_L(t)$$

As we can see, H is time dependent. And,

$$H(0) \approx h_L(0)$$

Also, $h_L(t)$ reduces to *0* after n time steps, so:

$$h_L(n) \approx 0$$

Finally, we have:

$$H(n) \approx h$$

This means, at the end of the annealing time, the energy function that we want to optimize, and which we represented by h, is recuperated.

Adiabatic Quantum Computation [9] uses a similar process to Quantum Annealing, but instead of using a time dependent local Hamiltonian, we use a time independent one h_L and we rewrite the total Hamiltonian as the following:

$$H\,(\mathrm{t}) = \frac{t}{n}h + (1 - \frac{t}{n})h_L$$

Similarly, we have:

$$H\,(0) = h_L$$

And,

$$H\,(n) = h$$

D-wave one implements such adiabatic process where qubits are set into "a state of quantum superposition, in which they're free to explore all [. . .] computational possibilities simultaneously, then you allow them to settle back into a classical state and become regular 1′s and 0′s again. The qubits naturally seek out the lowest possible energy state consistent with the requirements you specified in your algorithm back at the very beginning" [7]. In this regard, one can request the lowest value of a search space by considering it as the lowest energy state; thereafter the qubits will harvest all the search space and respond at the same time. Thus, the lowest point value is retrieved.

Quantum entanglement and superposition provide the system global information of the quantum process and its environment [11]. Similar to a conceptual Turing machine that has the whole infinite tape contents at its disposal in a specific moment in time, in contrast to the classical Turing machine which processes the tape square by square in step by step. This makes the system non-local, in other words not restricted to the locality principle as in a classical Turing machine [11]. In [11], Licata suggests that in order to benefit from the essential features of Quantum mechanics, which are quantum entanglement and superposition, in the design of a quantum-hyper-computing machine, we have to drop out the traditional universality characteristic of the machine. Thus, we should focus on "specific problem-oriented computation and based on its physical implementation" [11]. Such approach is referred as the "geometry of effective physical process... where computation is strongly linked to the very physical nature of the system and its global configuration, and the 'algorithm' is the evolution of the system itself in controlled experimental conditions" [11, 12]. An interesting point to mention is that the notion of programming is redefined and takes a whole new perspective because programming is now related to the geometry of the physical implementation [11, 12]. Last but not least, if an adequate configuration of an adiabatic quantum computer could be worked out, then it could "find the factors of a number that is the product of two large primes" [7] in a single night, which would take years for a conventional computer to solve; this will have enormous outcomes in exploiting nowadays encryption. Furthermore, D-wave one was able to find the lowest-energy configuration of a folded protein [10], which is a challenging problem in computational biophysics. This was "the first experimental and largest quantum annealing experiment related to an optimization problem in the physical sciences" [10 (Supplementary material)].

We have to note that Adiabatic Quantum Computation and Kieu Quantum Algorithm that we mentioned in the beginning of this section are examples of the "geometry of effective physical process" [12] approach that we discussed. For further investigation of this approach and its latest advancement, we refer the reader to [13] where the mathematical foundation of "Quantum Morphogenetic Computing" is presented as a general framework, which derives a non-Euclidean geometry of information from the probabilistic features of quantum phenomena (Superposition and Entanglement).

6 Conclusion

In 1999, Jack Copeland along with Diane ProudFoot [29] brought legacy to Alan Turing; by explaining and illustrating two well elaborated and forgotten ideas, which this brilliant scientist came up with, theorized and conceptualized. First, building artificial neural networks (e.g. an 'Unorganized Machine' which Alan Turing had suggested as a model of the unorganized neurons in an infant cortex that can be organized by suitable interfering training). Second, building a 'super computer' framework or a 'Hyper Machine' (e.g. The O-Machine; which is an augmented version of a Universal Turing Machine (UTM), theorized by Alan Turing as a UTM accompanied with a black box called the Oracle. The Oracle's function is to assist the UTM in deciding if a number is computable or not). Hypothetically, a Hyper Machine, if implemented, could bypass the limitations of the Turing Machine (i.e. The physical barrier of the Church-Turing Thesis).

Finally, when we discuss Hypercomputation, we should obviously mention the critic of Martin Davis towards the topic [30]. Martin Davis is skeptic about Hypercomputation research; he says that studying Hypercomputation is just a substitution of studying how to compute the non-computable. He claims that since we are finite human beings, who have a finite life span and who have access to finite data then it is impossible that we will be able to see the outcome of an infinite output that is computed by a Hypercomputer and to compare it with a Turing Machine, thus "no possible experiment could certify that a device is truly going beyond the Turing computable" [30]. Also, Martin Davis claims that if a device is to be considered as a Hypercomputer then it should be based on a physical theory that is absolutely correct. But, since any physical theory gives just an approximation to reality then it is impossible to have – to build - such a physical system that can have an absolute precision to infinity [30]. Furthermore, he says that Kieu's algorithm will not be able to find positive integer solutions for all Diophantine equations. Kieu responds to Martin Davis and other critics by saying that they just misunderstood his approach towards solving Hilbert's 10^{th} problem using the Quantum Adiabatic Theorem. Kieu says, "the noncomputability of Hilbert's tenth problem is the fact that we ask for a single finite procedure which can be applied to all the elements of sets of countably many Diophantine equations" [31]. But, he agrees that "that there is no single finite recursive algorithm for all Diophantine equations, but for each given equation we have to find a recursive algorithm anew each time" [31]. In fact, this is the essence of adiabatic quantum computing approach and the "geometry of effective physical process" [11, 12] approach, which are problem specific.

This review article highlighted on the advancements of three interesting examples of Hypercomputing models or Hyper Machines; Accelerated Turing Machines using Superluminal particles, Relativistic Computing based on recent discoveries in relativity theory and Quantum Computing using adiabatic quantum computer modeling and Quantum Morphogenetic Computing. The three examples were chosen because they are centered on new insights in quantum physics and relativity theory.

References

[1] Copeland, B.J. Hypercomputation (2002). *Minds and Machines Kluwer Academic Publishers.* 12, 461–502.

[2] Takaaki Musha. Possibility of Hypercomputation from the Standpoint of Superluminal Particles. *Theory and Applications of Mathematics & Computer Science* 3 (2) (2013) 120–128.

[3] Hajnal Andréka, István Németi, and Péter Németi (2009). General relativistic hypercomputing and foundation of mathematics. *Natural Computing 8(3):499-516.*

[4] Reynolds, C. C., Brenneman, L. W. and Garofalo, D., Black hole spin in AGN and GBHCs. Oct. 5, 2004. arXiv:astro-ph/0410116. (Evidence for rotating black holes.)

[5] Strohmayer, T. E., Discovery of a 450 HZ quasi-periodic oscillation from the mi-croquasar GRO J1655-40 with the Rossi X-Ray timing explorer. *The*

Astrophysical Journal, 553,1 (2001), pp. L49-L53. (Evidence for rotating black holes.) arXiv:astro- ph/0104487.

[6] Feynman, R.P., 1982, Simulating physics with computers. *Int. J. Theor. Phys.*, 21, 467.

[7] Lev Grossman, The Quantum Quest for a Revolutionary Computer, Time, Issue of 17 February 2014, Online Reference: http://time.com/4802/quantum-leap/

[8] Sergio Boixo, Troels F. Rønnow, Sergei V. Isakov, Zhihui Wang, David Wecker, Daniel A. Lidar, John M. Martinis and Matthias Troyer. Quantum annealing with more than one hundred qubits. *Quantum Physics* **arXiv:1304.4595v2**, arXiv.org, 2013

[9] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser, Quantum computation by adiabatic evolution, *quant-ph/0001106*

[10] Alejandro Perdomo-Ortiz, Neil Dickson, Marshall Drew-Brook, Geordie Rose and Alán Aspuru-Guzik. Finding low-energy conformations of lattice protein models by quantum annealing. *Nature.com, Scientific Reports* **2**, Article number: 571 (2012). doi:10.1038/srep00571

[11] I. Licata, Beyond Turing: Hypercomputation and Quantum Morphogenesis. *Asia Pacific Mathematics Newsletter*, Volume 2, Issue 3, 2012, http://www.asiapacific-mathnews.com/02/0203/0020_0024.pdf

[12] I. Licata, Effective physical processes and active information in quantum computing, *Quantum Biosystems* **1** (2007) 51–65, http://arxiv.org/pdf/0705.1173.

[13] I. Licata and G. Resconi, Information as environment changings. Classical and quantum morphic computation, in M*ethods, Models, Simulation and Approaches Toward A General Theory of Change*, eds. G. Minati, *et al.* (World Scienti c, 2012), pp. 47–81.

[14] Wiles, A. "Modular Elliptic-Curves and Fermat's Last Theorem." *Ann. Math.* 141, 443-551, 1995.

[15] Kenneth Ribet in an interview with The New York Times by Gina Kolata. "At Last, Shout of 'Eureka!' In Age-Old Math Mystery", *The New York Times*, June 24, 1993.

[16] Earman,J. and Norton, J., Foreverisaday: supertasks in Pitowsky and Malament-Hogarth spacetimes. *Philosophy of Science* 60 (1993), 22-42.

[17] Mandelshtam, Tamm. The uncertainty relation between energy and time in nonrelativistic quantum mechanics, *J. Phys. (USSR)* 9 (1945), 249-254.

[18] Hilgevoord, Jan and Uffink, Jos, "The Uncertainty Principle", *The Stanford Encyclopedia of Philosophy* (Spring 2014 Edition), Edward N. Zalta (ed.), URL = <http://plato.stanford.edu/archives/spr2014/entries/qt-uncertainty/

[19] Lloyd, S. (2000). Ultimate physical limits to computation. *Nature* 406, 1047–1054.

[20] Recami, E. (2001). Superluminal motions ? a bird's-eye view of the experimental situation. *Foundations of Physics* 31(7), 1119–1135.

[21] Brown, J. (1995). Faster than the speed of light. *New Scientist* 01 April 1995, 26–30.

[22] Steinberg, A. M., P. G. Kwiat and R. Y. Chiao (1993). Measurement of the single-photon tunneling time. *Phys. Rev. Lett.* 71, 708–711.

[23] Wang, L. J., A. Kuzmich and A. Dogariu (2000). Gain-assisted superluminal light propagation. *Nature* (6793), 277– 279.

[24] Etesi, G. and N emeti, I. (2002). Turing computability and Malament-Hogarth spacetimes. *International Journal of Theoretical Physics 41(2)*: 342-370.

[25] Nemeti, I. and David, Gy. (2006). Relativistic computers and the Turing barrier. *Journal of Applied Mathematics and Computation* 178, 118-142.

[26] Kieu T. D. (2003) Computing the noncomputable. *Contemporay Physics* 44(1): 51-71.

[27] Gödel, K., 1951, in a statement delivered to *The American Mathematical Society.*

[28] Book: The Fabric of Reality by David Deutsch. New York: Penguin Books, 1997 Chapter 9.

[29] B. Jack Copeland and Diane Proudfoot. Alan Turing's Forgotten Ideas in Computer Science, *Scientific American,* April 1999

[30] Martin Davis. Why there is no such discipline as hypercomputation. *Applied Mathematics and Computation* 178(1): 4-7 (2006)

[31] T.D. Kieu, Reply to Andrew Hodges, *Quantum Physics* arXiv:quant-ph/0602214v2, arXiv.org, 2006