

# Classical Computer, Quantum Computer, and the Gödel's theorem

Biao Wu\*

*International Center for Quantum Materials,  
School of Physics, Peking University, Beijing 100871, China  
Wilczek Quantum Center, School of Physics and Astronomy,  
Shanghai Jiao Tong University, Shanghai 200240, China and  
Collaborative Innovation Center of Quantum Matter, Beijing 100871, China*

(Dated: April 30, 2022)

## Abstract

I show that the cloneability of information is the key difference between classical computer and quantum computer. As information stored and processed by neurons is cloneable, brain (human or non-human) is a classical computer. Penrose argued with the Gödel theorem that human brain is not classical. I demonstrate with an example why his argument is flawed. At the end, I discuss how to go beyond quantum computer.

In the book *Frank Wilczek, 50 years of theoretical physics* (World Scientific, 2022), pp.281-290; [https://doi.org/10.1142/9789811251948\\_0021](https://doi.org/10.1142/9789811251948_0021); arXiv:2106.05189

---

\*Electronic address: [wubiao@pku.edu.cn](mailto:wubiao@pku.edu.cn)

Galileo wrote in 1623, “Philosophy is written in this grand book – I mean the universe – which stands continually open to our gaze, but it cannot be understood unless one first learns to comprehend the language in which it is written. It is written in the language of mathematics, and its characters are triangles, circles, and other geometric figures, without which it is humanly impossible to understand a single word of it; without these, one is wandering about in a dark labyrinth.” [1] This was a remarkably bold and visionary statement at the time when there were only a few accurate mathematical descriptions of physical phenomena, such as free fall and pendulum. Currently, there is no doubt that the universe is written in the language of mathematics. A physical process is literally a demonstration of a mathematical solution. As computation is a process of finding mathematical solutions, a physical process can be regarded as a computation of a mathematical equation. Or simply, physics is computational. How about the reverse, is computation physical? The answer had been “No” until the discovery of quantum computers [2–5]. It is now widely accepted that computation is physical. Different models of computation correspond to different physical laws. Classical physics underlies Turing machine and other models of classical computers, quantum physics is behind quantum computers.

However, it is not obvious at all why a classical computer is classical and a quantum computer is quantum. The models of classical computers do not operate according to the Newton’s equations of motion; neither do the models of quantum computers run according to the Schrödinger equation. There is even no Planck constant  $\hbar$  in theoretical models of quantum computers. It turns out that being classical and being quantum are concepts more fundamental than equations of motion. The nature of a computer is determined by the cloneability of the information processed by the computer. If the information is cloneable, the computer is classical; if it is uncloneable, the computer is quantum. As the information processed in neurons are cloneable, brains (human or non-human) are classical computers. Penrose argued that the Gödel theorem implies that human brains (not all brains) are not classical [6]. I shall show that this is due to his misunderstanding of the theorem. The Gödel theorem in fact implies that there are innumerable mathematical statements so that human mathematicians, who have only finite physical resources, cannot prove all of them to be true or false. An example is constructed to demonstrate it. In the end, I shall give an example on how to construct theoretical models of computing that are more powerful than quantum computers [7]. To do this, the information processed by the computers has to be

different from both classical computers and quantum computers.

## I. CLASSICAL COMPUTER AND QUANTUM COMPUTER

There are many different models of classical computers, such as the famed Turing machine and the conventional digital computer. They are all equivalent in terms of theoretical computing power [8]. I shall use the reversible classical computer to illustrate the essential features of a classical computer. The primary reason is that quantum computers can be viewed as a straight-forward extension or quantization of reversible classical computers, which makes the comparison between classical computer and quantum computer more transparent.

A reversible classical computer has one universal logic gate, the Toffoli gate, along with an array of bits. Each bit has only two values, 0 and 1. The Toffoli gate, as shown in Fig. 1(b), has two control bits and one target bit. Only when both control bits are 1, the target bit switches between 0 and 1 [8].

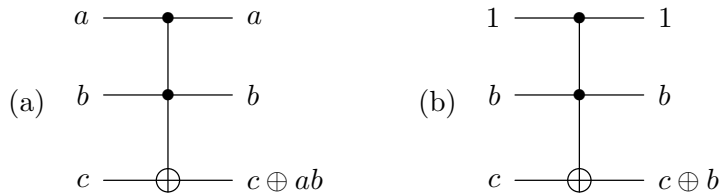


FIG. 1: (a) the Toffoli gate; (b) CNOT gate constructed from the Toffoli gate. The inputs  $a, b, c$  are Boolean variables.

On top of the reversible classical computer, if we add two more logic gates and replace bits with qubits, we have the standard circuit model of quantum computer. The two additional gates are the Hadamard gate  $H$  and  $\pi/8$  gate  $T$  [8],

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \quad T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}. \quad (1)$$

The usual universal quantum gate, CNOT gate, can be easily constructed from the Toffoli gate as shown Fig.1(b). The key difference between qubit and bit is that a qubit can be in a superposition state

$$|\psi\rangle = \alpha |0\rangle + \beta |1\rangle. \quad (2)$$

The states of a bit correspond to two special cases,  $\alpha = 1, \beta = 0$  and  $\alpha = 0, \beta = 1$ .

The Hadamard gate is the key in quantum computer. It is the only universal gate that is capable of generating a superposition state out of a non-superposition state, for example,

$$H |0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle). \quad (3)$$

Only with Hadamard gates will we be able to generate more superposition terms, change the superposition amplitudes, and facilitate interference in the course of computation. This shows that quantum computer can be viewed as a quantization of reversible classical computer. The quantization is achieved primarily with the Hadamard gate and additionally with  $\pi/8$  gate  $T$ , but remarkably without Planck constant. A classical computer with  $n$  bits have exactly  $2^n$  states,

$$|J\rangle = |J_1, J_2, \dots, J_k, \dots, J_n\rangle \quad (4)$$

where  $J_k$  is the  $k$ th binary digit of  $J$  ( $J = 0, 1, 2, \dots, 2^n - 1$ ). In contrast, a quantum computer with  $n$  qubits has infinite states that live in a Hilbert space spanned by  $|J\rangle$ 's. In general, a quantum computer is in a superposition state

$$|\Psi\rangle = \sum_{J=0}^{2^n-1} a_J |J\rangle. \quad (5)$$

## II. CLONEABLE STATES AND UNCLONEABLE STATES

The reversible classical computer is clearly not quantum as it does not allow generation of superposition. But why is it classical? Why is Turing machine classical? The conventional digital computer is also regarded as classical even it is irreversible. What do these models of classical computers have in common with classical mechanics? Similarly, why is the model of quantum computers described above quantal? There is even no Planck constant in the model.

To answer these questions, let us review dynamical systems in general. A dynamical system has a configuration space and an evolution rule (or a set of evolution rules). Its state is a point in the configuration space. The system can start at any point in the configuration space. The evolution rule dictates it to evolve deterministically from one state to another. Classical mechanics is a dynamical system. The configuration space is the phase space and the evolution rule is the Newton's equations of motion. Quantum mechanics is a dynamical system. The configuration space is the Hilbert space and the evolution rule is

the Schrödinger equation. Mathematically, classical mechanics and quantum mechanics are essentially the same type of dynamical systems. Koopman and von Neumann formulated classical mechanics mathematically as quantum systems [9, 10]. And Heslot converted the Schrödinger equation mathematically into a set of Newtons equations of motion [11].

This shows that the equations of motion are not what makes classical mechanics classical and quantum mechanics quantal. The key difference is how their states, the points in their configuration spaces, are related to physical reality. The coordinates of a point in the phase space are momenta and positions, which can be directly measured and cloned. In contrast, a point in a Hilbert space, usually called wave function, is not directly observable and can not be cloned in general. To relate a wave function to physical reality, one has to first select a complete set of orthonormal basis, which are usually the eigenfunctions of a physical quantity such as momentum, position or angular momentum. This selection of basis is outside of the configuration space and the equations of motion, and is done independently. When the wave function is expanded in this basis, the expansion coefficients (or the coordinates) are not measurable, and thus not cloneable. Upon measurement, the basis vector is observed its corresponding eigenvalue, not the coefficients (or the coordinates). To distinguish this crucial difference, we call classical state (a point in phase space) *cloneable* and quantum state (a vector in Hilbert space) *uncloneable*. It is this cloneability of their states that separates a classical system from a quantum system, not its equations of motion. This cloneability is related to the well-known no-cloning theorem for quantum states [12, 13]. According to this theorem, all the states  $|J\rangle$  of bits can be cloned but not their superpositions, such as the states in Eq. (5).

The states of a classical particle are cloneable. Its position and momentum can be measured and recorded directly, ultimately cloned faithfully with negligible errors. The states  $|J\rangle$  of bits are cloneable not only theoretically but also in practice: the two states, 0 or 1, of a bit, often represented by different voltages, can be measured and observed directly, and cloned faithfully. It is this common feature with a classical particle that makes bits classical. This is the key reason that any computer with bits, such as the Turing machine, the conventional digital computer, and the reversible computer are classical. In contrast, the states of qubits are vectors in the Hilbert space just like the wave functions in quantum mechanics. Upon measurement, only partial information of these states can be recorded; they can not be cloned faithfully with negligible errors. As a result, computers

with qubits are quantum computers. It is now clear. Classical computer is classical because its states (or information) are cloneable. Quantum computer is quantal because its states (or information) are uncloneable. One can force the quantum states to be cloneable. This is exactly what happens when we simulate a quantum system on a classical computer. The price to pay is that the memory size of a classical computer would have to grow exponentially with the size of the quantum system. It is precisely what motivated the concept of quantum computers [2, 3]. Note that one can also repeat measurements on a given quantum state and ultimately clone it faithfully. Unfortunately, the number of measurements needed grows exponentially with the system size [14].

### III. BRAIN AND GÖDEL'S THEOREM

The brain (human or non-human) is clearly a classical computer because the states of each neuron are cloneable. A neuron state is marked by electrical pulses (usually called action potential) passing through the axon, opening of ion channels, sodium and potassium concentrations, activation of chemical or electric signals at synapses. These physical and chemical states can be measured and recorded directly and cloned faithfully, at least in principle. Technically, it is challenging to clone faithfully the functions of a single neuron, not to mention a network of neurons (brain). But it is an issue of technology, not an issue of theoretical principles. Technically, we are still far away from building a machine that moves in a jungle as adeptly as monkeys. Nevertheless, the monkey's brain, muscle, and other body parts function classically.

That human brain is a classical computer is also evident in the process of a mathematician proving a theorem. When a mathematician proves a theorem, axioms or assumptions are inputs, mathematical inductions are logical gates, known theorems or lemmas are sub-routines. A completed proof is essentially a classical algorithm because all the symbols and induction rules can be written down without any ambiguity, and be cloned faithfully from mathematician to mathematician. In other words, they are cloneable just like the states of bits.

However, Penrose argued that human brain is not a classical computer and may be of quantum nature [6]. He believes that this is implied in Gödel's incompleteness theorem, which states that there are statements in any consistent formal system  $F$  which can neither

be proved nor disproved in  $F$ . Let me first summarize Penrose's arguments or proof [6]. He listed all the computations  $C_q(n)$  that take a natural number  $n$  as its input.  $C_q(n)$  is the  $q$ th algorithm that computes on the input  $n$ .  $A(q, n)$  of two inputs  $q$  and  $n$  is an algorithm that intends to ascertain whether  $C_q(n)$  will stop: if  $A(q, n)$  stops,  $C_q(n)$  does not stop. With Cantor's diagonalization argument, Penrose arrives at the statement (M): if  $C_k(k)$  stops,  $C_k(k)$  does not stop. This leads him to conclude that

Human mathematicians are not using a knowably sound algorithm in order to ascertain mathematical truths.

This means that human brain is not a classical computer. For me, (M) is simply a self-contradictory statement and means that at least one of the assumptions made at the start of Penrose's proof is wrong. In fact, Penrose implicitly assumed that all the algorithms  $C_q(n)$  are countable and therefore can be listed. This assumption is certainly wrong. In Cantor's proof that the real numbers are innumerable, he first assumed that the real numbers are countable and can be listed. With the diagonalization argument, he found a real number outside of the list. A contradiction so that the assumption that the real numbers are countable is wrong. Cantor's proof also implies that irrational numbers can not be exhausted by a finite number of different operations with finite symbols. In contrast, all rational numbers can be obtained by one operation (division) with 10 symbols (0,1,2,3,4,5,6,7,8,9).

Similarly, it is wrong to assume that algorithms are countable. Here is an example that demonstrates that there are innumerable algorithms. A number is said embedded in  $\pi$  if its decimal digits can be found in  $\pi$  in the sequential order. For example, 1.23456 is embedded in the first 50 decimal digits of  $\pi$ ,

$$3.\underline{1}4159\underline{2}65\underline{3}589793238\underline{4}62643383279\underline{5}028841971\underline{6}939937510\dots \quad (6)$$

$R^{(n)}(x)$  is an operation that repeats  $n$  times each decimal digit of  $x$ . For example,  $R^{(1)}(1.23456) = 1.23456$  and  $R^{(3)}(1.23456) = 111.222333444555666$ . For every real number  $x$ , we construct a list of mathematical statements  $E^{(n)}(x)$ :  $R^{(n)}(x)$  is embedded in  $\pi$ . The statement  $E^{(1)}(\sqrt{2})$  is probably true. Its first twelve decimal digits are embedded in the first 100 digits of  $\pi$ .

$$3.\underline{141}592653589793238\underline{462}643383279502884\underline{1971693}9937\underline{510}582097494459230781\underline{64062}862089986280\underline{348}2534211\underline{706}79\dots \quad (7)$$

Its thirteenth digit is 3 and will be found in the rest digits of  $\pi$ , similarly its fourteenth digit 0, and so on. The statement  $E^{(2)}(\sqrt{2})$  is no longer as obvious. Here is the situation for the first 200 digits of  $\pi$ .

$$\begin{aligned}
&3.14159265358979323846264338327950288419716939937510 \\
&5820974944592307816406286208998628034825342\underline{11}70679 \\
&821480865132823066470938\underline{44}609550582231725359408128 \\
&48\underline{11}74502841027019385211055596\underline{446229}4895493038196\dots
\end{aligned} \tag{8}$$

But still it is hard to believe that the next two digits 11 never appear in the rest digits of  $\pi$ , similarly 33, and so on.  $E^{(2)}(\sqrt{2})$  is probably true. As  $n$  increases, we become less confident that the statement  $E^{(n)}(\sqrt{2})$  is true. Consider  $n = 6$ . The first string of six or more consecutive identical digits is six 9's starting at the 762nd decimal place of  $\pi$ , the next is again six 9's starting at position 193034, and the next is six 8's starting at position 222,299 [15].

To verify whether  $E^{(n)}(x)$ 's are true or not, one can set up algorithms that simply compute every decimal digits of  $x$  and  $\pi$  and check them one by one. This method can show that some statements, such as  $E^{(6)}(9)$ , are true. When  $x$  is irrational, this method fails. It can not even show with 100% certainty whether  $E^{(1)}(\sqrt{2})$  is true because mathematicians (including machines) have only finite resources and they can not compute infinite number of digits. However, sometimes one does not need to compute out all the digits of an irrational number to verify a statement true or false. For example, one can show that  $\sqrt{2}$  is irrational and  $E^{(1)}(\pi)$  is true without computing out all the digits. So, it is possible that mathematicians can figure out a 'clever' way or algorithm to verify that  $E^{(1)}(\sqrt{2})$  is true. However, it is hard to imagine that 'clever' algorithms can be found for all irrational numbers and any  $n$ .

As the real numbers are innumerable, we have innumerable mathematical statements  $E^{(n)}(x)$ . Each statement  $E^{(n)}(x)$  is clearly also an algorithm with two inputs  $n$  and  $x$ , and can be identified with Penrose's  $C_q(n)$  via  $q = x$ . This example shows that Penrose's assumption that  $C_q(n)$ 's are countable is wrong.

We now face two realities: (1) there are innumerable number of mathematical statements; (2) human mathematicians are only capable of using finite number of symbols, finite sets of induction rules, and doing finite steps of deductions ( similar to a finite number of logical gates in an algorithm). The latter means that the mathematical statements that can be



verified by human mathematicians are countable and can be listed. Therefore, it is clear that mathematicians can not prove all the mathematical statements, which are innumerable, true or false. This is the essence of Gödel's theorem. Therefore, Gödel's theorem has nothing to do with whether human brain is classical or not. The theorem just shows with mathematical rigor a common sense: it is impossible for any thinking machine (including human brain) with finite resources to solve innumerable problems.

Gödel's theorem is often explained with self-referencing examples, for example, this statement is not true. This self-referencing is made rigorous in its proof with the diagonalization argument. It gives the theorem a mystique appearance that obscures its essence and true implication: innumerable problems can not be solved with finite physical resources.

There are obvious differences between brain and the conventional digital computer. It is a mundane task for many animals, including human, to walk or run on difficult terrains; it is declared a major achievement whenever robots's walking abilities have been improved in one or another aspect. It takes a long time for a human to add two large numbers; it takes almost no time for a digital computer to do it. This does not imply that human brain may be a quantum computer. It is simply that we do not understand the networking of neurons inside human brain. Just as we do not need to know every technical details of a digital computer to know for sure that it is classical, we can be absolutely sure that human brain is classical before we understand it completely.

Galileo said that physics is written in the language of mathematics; Gödel proved that mathematics is limited by physics. After all, human is physical. All human products, including mathematical theorems, are limited by the physical laws. What we know and will know is negligible relative to what we will never know. This is in a sense good as we will never run out of truths to discover.

#### **IV. BEYOND QUANTUM COMPUTING**

Is it possible to go beyond quantum computer? According the analysis in Section II, it is impossible in practice since we do not have any experimentally tested theory that goes beyond quantum mechanics. Theoretically, it is certainly possible. We only need a dynamical system whose states in its configuration space are related to physical reality in a different way from both classical mechanics and quantum mechanics.

One of such systems was discussed long time ago by Pauli [16] who was inspired by Dirac [17]. It was considered independently many years later as a generalization of Bogoliubov-de Gennes equation [18]. It was named Lorentz quantum mechanics because its dynamical evolution is generalized Lorentz transformation. The key features of the Lorentz mechanics are: (1) its states are vectors in a complex linear space with an indefinite metric; (2) its dynamics is generalized Lorentz transformation; (3) there are two sets of states, one with positive inner products and the other with negative products. The third feature is crucial: the positive states are regarded as physical and can be measured; the negative one are regarded unphysical. This means that the states in the configuration space, an inner product space with indefinite metric, are not only uncloneable and half of them are also unphysical. This is different from both classical states and quantum states.

We have proposed a new theoretical model of computing based on this Lorentz quantum mechanics [7]. We find that the search algorithm designed for this new computer model is exponentially faster than the Grover's algorithm [19]. It shows explicitly that one can go beyond conventional quantum computer. In a new theory that unifies quantum theory and gravity, if its states are different from both classical states and quantum states, we would also expect to have a more powerful model of computing.

There are two computer models that were shown to be more powerful than quantum computer [20, 21]. Both add nonlinearity to quantum mechanics and claim that nonlinearity is behind the enhancement of computing power. It is suspicious argument. Classical mechanics is inherently nonlinear and no one has been able to exploit its nonlinearity to enhance computing power. It is no surprise that both models were critically questioned from the theoretical point of view[22].

\*\*\*\*\*

**Note:** I started looking seriously into quantum computing around 2015. As usual for a newcomer in the field, I soon had a “breakthrough”, a constant-time search algorithm; as an experienced researcher, I also felt uncomfortable. Something may be wrong, but I was not sure what it was. In the summer of 2015, I visited Stockholm, where I discussed this work with Frank. Soon it was clear what the mistake was. What was inspiring was that Frank kept discussing with me on this problem, suggesting various ways to amend the mistake. The discussion continued even after I left Stockholm. The eventual conclusion was that

my original proposal was hopeless. Nevertheless, I was encouraged and started to try new approaches. I discussed them with Frank when he visited Hangzhou in the fall of 2015. Again they were flawed. But this time Frank came up with a beautiful idea, using resonance to do Grover search. About one month later, Frank sent me a note where he had worked out all the details of this idea with an additional interesting ingredient, monitor qubit. It eventually led to a joint paper [23]. More collaboration between Frank and my group followed. Recently, we had a wonderful quantum algorithm, which significantly outperforms its classical counterpart, for approximating maximum independent sets [24, 25]. It has been wonderful and inspiring experience. I learned from Frank this life lesson: *mistakes may be embarrassing, but often they are opportunities for progress.*

---

- [1] G. Galilei, *The Assayer* (University of Pennsylvania Press, 1960).
- [2] Y. I. Manin, *Computable and Noncomputable* (Sov. Radio., 1980).
- [3] R. P. Feynman, International Journal of Theoretical Physics **21**, 467 (1982).
- [4] D. Deutsch, Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences **400**, 97 (1985).
- [5] D. Deutsch, Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences **425**, 73 (1989).
- [6] R. Penrose, *Shadows of the Mind* (Oxford University Press, New York, 1994).
- [7] W. He, Z. Wang, and B. Wu, arXiv:2103.10315 (2021).
- [8] M. A. Nielsen and I. L. Chuang, *Quantum computing and quantum information* (Cambridge University Press, Cambridge, 2000).
- [9] B. O. Koopman, Proceedings of the National Academy of Sciences of the United States of America **17**, 315 (1931).
- [10] J. von Neumann, Annals of Mathematics **33**, 587 (1932).
- [11] A. Heslot, Phys. Rev. D **31**, 1341 (1985).
- [12] J. L. Park, Foundations of Physics **1**, 23 (1970).
- [13] W. K. Wootters and W. H. Zurek, Nature **299**, 802 (1982).
- [14] H.-Y. Hu and Y.-Z. You, arXiv:2102.10132 (2021).
- [15] D. Andersen, *The pi searcher*, URL <https://www.angio.net/pi/bigpi.cgi>.

- [16] W. Pauli, *Rev. Mod. Phys.* **15**, 175 (1943).
- [17] P. A. M. Dirac, *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences* **180**, 1 (1942).
- [18] Q. Zhang and B. Wu, *New Journal of Physics* **20**, 013024 (2018).
- [19] L. K. Grover, *Phys. Rev. Lett.* **79**, 325 (1997).
- [20] D. S. Abrams and S. Lloyd, *Physical Review Letters* **81**, 3992 (1998).
- [21] D. Bacon, *Physical Review A* **70**, 032309 (2004).
- [22] C. H. Bennett, D. Leung, G. Smith, and J. A. Smolin, *Phys. Rev. Lett.* **103**, 170502 (2009).
- [23] F. Wilczek, H.-Y. Hu, and B. Wu, *Chinese Physics Letters* **37**, 050304 (2020).
- [24] B. Wu, H. Yu, and F. Wilczek, *Phys. Rev. A* **101**, 012318 (2020).
- [25] H. Yu, F. Wilczek, and B. Wu, *Chinese Physics Letters* **38**, 030304 (2021).