Shafiq *LGURJCSIT* 2018

LGURJCSIT
ISSN: 2519-7991

LGU R Journal for
Computer Science IT

Research Article

Vol. 2 issue 1 Jan. March 2018

Quantum Limits, Computational Complexity and Philosophy – A Review

Shamaila Shafiq
shamailashafiq@gmail.com
Lahore Garrison University, Lahore, Pakistan

Abstract:

Quantum computing physics uses quantum qubits (or bits), for computer's memory or processor. They can perform certain calculations much faster than a normal computer. The quantum computers have some limitations due to which the problems belonging to NP- Complete are not solved efficiently. This paper covers effective quantum algorithm for solving NP-Complete problems through some features of complexity theory, that we can simplify some of the philosophical interest problems.

Keywords: Computational Complexity, Quantum Computation, Qubits, Complexity Theory, Philosophy.

1. Introduction

In theoretical computer science and mathematics, computational complexity is the theory of the branch of computation which classifies the problem by connecting classes to each other according to their inherent difficulty.

Computability theory coined by Alan Turing, Kurt Gödel, Alonzo Church, and more in 1930's has significantly inclined to philosophy, logic and artificial intelligence.

NP-Complete issues has been raised by the computational complexity theory similar to the open key cryptography, deductive contention from a scientific proclamation and the hypothetical points of interest for quantum computation and machine learning.

The theory of computation suggest that if something is quantifiable, computational complexity can be achieved in a restricted time or it will take longer than the life expectancy of the universe [2] as it interests the computer minds but not the philosopher's.

Philosophy is primarily being inquisitive of assumptions, those axioms that form the first point for any mathematical or scientific insight and complexity theory relates to it in some of the philosophical disciplines.

The key characteristic of quantum computers is that it uses qubits instead of bits. A qubit may be a particle, for example, an electron, through turn up signifying 1, turn down signifying 0, and states of quantum named as superpositions that contain turn up plus turn down at one time [5].

Before preceeding further a few important complexity classes are defined below. There are three broad overlapping categories of the complexity classes by the computer experts as per how much computational steps it takes by the best-known algorithms.

1.1 P Problems;

P means "Polynomial". It comprises of the decision problems for which an efficient algorithm exists [7, 14]. The problems solvable by a Turing machine in polynomial time fall in this class. For example: Is there any single town accessible from each other on a map? Another example is whether a number is prime or composite.

1.2 NP Problems;

NP means "Nondeterministic Polynomial Time". It includes all those issues that can be perceived as right in polynomial time [7]. The problems solved by nondeterministic Turing machine to make a guess of value of the permit, certifiable in polynomial time are called as NP problems. For example: Finding out the prime factors of n digit number that is a product of two big prime numbers.

1.3 NP-Complete Problems;

NP-Complete Problem is the hardest of all problems. If for any of the NP problems, an efficient algorithm is found then it could be modified towards resolving great challenges. It was a concept which was initiated by Stephen A. Cook, Richard Karp and Leonid Levin. The Problem of defining the way to place n boxes of different size(s) in a trunk of definite size lies in NP-Complete Problem. More examples can include the Sudoku game and jigsaw puzzles. A well-known problem can be of the Traveling Salesperson Problem [7].

The paper will discuss the debate on Philosophy versus Complexity Theory, Importance of Polynomial Time, Computational Complexity and the Turing Test, Logical Omniscience problem, Quantum Computing (Including Limits of Quantum), Fusion of Complexity, Space, and Time including the CTC (Closed Timelike Curves), Disapproval of Complexity Theory, Future Ways and Conclusions .

2. Philosophy Versus Complexity Theory

It would be astonishing that complexity theory has no philosophical touch similar to the computability theory, because computability theory has math in it. Complexity theory have rich philosophical associations after World War II; computer science theory move towards the technology turn and lost its connection with the heritage of the philosophy.

Therefore by determining some features of complexity theory, as well as some philosophical problems, complexity view can streamline and right the communication gap between philosophy and complexity theory.

It would not be incorrect to say that complexity theory has closer association with the sciences. It compels to think us about quantum physics, evolution, statistical physics, human language getting hold of that would be of no meaning from a computability viewpoint. Complexity can be distinguished from computability in the variety of mathematical methods used. Mostly the mathematical logic primarily comes from complexity (like computability); today it appeals on arithmetic, probability, abstract algebraic structures theory, combinatorics, Fourier analysis and almost every known topic.

According to the theoretical computer researchers, an algorithm that takes upper bound polynomial function running time (n) is considered more efficient than lower bound exponential function running time like $2^{n.}$

However; the thorough efficiency of an algorithm may be measured by the computation model and the closure properties of the polynomial and exponential time.

Practically efficient algorithms are those that take polynomial time to solve the problems and inefficient are the ones which take exponential time as its obvious for difference that 1.0000001^{n} running time algorithm will be faster than an algorithm that takes $2^{1000000}$ running time.

3. Importance of Polynomial Time

The importance of polynomial/exponential time cannot be overlooked. Its significance in biology, mathematics, and science has been enlightened with the examples below:

3.1 Entscheidungs Problem

In 1920, David Hilbert posed a challenge or more suitably a dream called “The Entscheidungs problem”. The problem inquires for an algorithm that takes a mathematical statement and description of a formal language as an input and yields an output as either "True" or "False". However, in 1930’s the work of Gödel, Turing and Church destroyed Hilbert’s dream.

These results have impacted greatly on the philosophy of mathematics and logic. There have been attempts to apply the results also in other type of philosophy such as the philosophy of mind.

In theoretical computer, Gödel’s letter to John von Neumann [6] in 1956 got famous since its rediscovery in the 1980’s. Given a formal system F, consider the problem of determining whether a mathematical statement S has evidence in F having n symbols or less. This “shortened Entscheidungs problem” is clearly decidable as contrast to Hilbert’s original problem.

This problem seems to be an NP-Complete problem rather than NP problem. Whether P ≠ NP is same as enquiring NP-complete problem as it determins polynomial time, and is similarly equal to questioning all of the problems. The Gödel says that if P = NP then if a theorem has a proof of rational size, it can be proved in rational time. For that one might say that "for all practical purposes," Hilbert's tried to overcome mechanizing the mathematics, in spite of the undecidability properties of Gödel, Turing, and Church. If one agrees then it seems to say that until P versus NP is not resolved the story of Hilbert's Entscheidungs problem—its growth, its descent, and philosophy debate will not be over.

3.2 Advancement:

In 1972, Kurt Gödel carried his own uncertainties about evolution in a letter to Hao Wang [3].

The letter of Gödel's to Wang may assume to forestall current effort by the computer researcher creativity compared to other types of areas in the group-project. In order to improve the performance of students in these two areas, one may think of enhancing the way practical sessions are run, increase the number of sessions in these areas, use techniques such as debates [10].

Leslie Valiant, for building a measurable "theory of evolvability" [6].

If we presume that the Gödel was correct, then the systematic worldview of recent biology was computational irregularity hypothesis appears to be gloomily a long way from having the capability to demonstrate anything of the kind. For now, people have been thinking deeply about that and identified enormous difficulties of proving even such "obvious" and the conjectures like P = NP.

3.3 Identified Integers:

The idea of "knowledge" in mathematics concerns the philosophical importance of the polynomial and exponential. As of 7th January 2017, the largest prime number as stated by the GIMPS is $2^{74,207,281-1}$, a number having 22,338,618 digits [1]. If p is identified then it means that we can simply take print out its decimal digits. In fact, it is beyond the earth capacity to print out the prime number decimal digits or if we assuming storing it in the computer's memory.

If we identify an algorithm by taking positive integer k as input and outputs the decimal characters of $p = 2^k - 1$ using the polynomial number of operation. On the other hand, the core is that any efficient algorithm is not known which is alike to that which gives the first prime number larger than $2^k - 1$.

Beyond the benefit that it provides to theoretic computer science, it can be a rich source for philosophy as far as the polynomial/exponential study is concerned.

4. Computational Complexities and the Turing Test

Alan Turing in 1950, proposed a Turing test in which the intelligence of human and computer becomes indistinguishablet. So it poses a question that "Can computers can reason like human being?" It blends up two issues: one is related to metaphysics and the other a "practical" issue. Firstly, if a computer passed a Turing test then it would not be wrong to assign them the "consciousness," "qualia," "intentionality," "prejudice," "personhood," "bigoted" or whatever other enchanted position we wish to offer to the other humans? Secondly can such a program which can pass a Turing test can be written in reality?

People, who considers Artificial Intelligence as a metaphysical option, have not gone through experiments before considering it and thus here we can say that people can get advantage from philosophy.

John Lucas opposed and Roger Penrose in the book, *The Emperor's New Mind and Shadows of the Mind* [8], expanded that as per the Incompleteness Theorem, one thing that a computer making inferences through particular formal rules can never "see" its own rules stability [4]. Therefore humans can never be simulated by machines. If one believes that the brain itself is basically a well-organized standard Turing machine, then one can have a normal explanation for the reason no one has ever discovered that such machines can never simulate a human brain.

The advancement in quantum computation is moving towards the Quantum Robots as a quantum computer which can be described as a quantum Turing machine [11] but whether the speedup for performing parallel computations or tasking as in case of Shor's, remains to be understood [9].

4.1 Can Humans Solve NP-Complete Problems in Less Time?

Though it is impossible to underrate the human intelligence, but if we compare the human beings intelligence with the computer's memory that far how much better is the human brain at solving the problems like of NP-Completeness.

If we take an example for which human beings are good at are the search problems for instance of high-level structure or semantics or ironically designing genius computer algorithms even if computers as compared to humans, were sound at factoring large numbers. Certainly, in some areas such as puzzle-solving, for which computers can inspect the solutions loads of times faster, but if we see now humans are much better at making either the results trivial or even impossible if we talk about finding the global patterns or solving the regularities in a puzzle. Hence, in general, the human's intelligence can never be overcome by the so-called machines or computers up to the times to come.

5. Logical Knowledge Problem

Normally, formal descriptions of knowledge include customary "logical" axioms as follows:

- If you distinguish R and S, it distinguishes R & S.
- If you distinguish R, then you also distinguish that you distinguish R.

- If you don’t distinguish R, then you know that you don’t distinguish R.

Now, to some extent we can state what Jakko Hintikka named the logical omniscience problem.

An example for illustration: Can we think that a normal three to four-year-old kid know that adding two numbers that are real is commutative or not? If we try to tell that kid in the above-mentioned way then surely, he will not understand. However, if we show that child the pile of blocks and tell him to make that pile high by shuffling the blocks, he maybe wouldn’t make incorrectness that involved visualizing that addition was non-commutative.

The example strongly propose that only a slight portion of anything that we mean by “knowledge” is the knowledge concerning the truth or untruth of one’s propositions.All of the above questions could be inferred as asking: Do we have an algorithm that can solve large group of queries of some form?

The logical omniscience problem has not yet been able to be solved by computational complexity theory in the intellect of giving a sufficient recognized reason of knowledge that also shuns building illogical guesses.

6. Quantum Computing

Quantum computing is an idea for quantum mechanics that can resolve certain computational problems much earlier than how we find to resolve them today [11,13,5]. For that one need to construct a new sort of computer, knowing of using the quantum effects of superposition and interference. It is a huge challenge to form such a large computer that’s solves interesting problems for engineering and physics.

A polynomial-time quantum algorithm was presented by Peter Shor in 1994, for factoring integers, and as a consequence breaking maximum of the cryptographic codes being used on the internet nowadays mostly for financial transactions.

Shor’s algorithm also gave indication that converting from classical to quantum computers will increase the discussion of problems which are solved in polynomial-time [12, 15].

If we actually could construct a mystic computer proficient of answering an NP-complete problem in an instant, making the world a very different place. The magical computer could solve the problems like the following:

- Could look for whatsoever forms might occur in data of stock-market or the brain activity or the weather records.

- Could also automate mathematical creativity.

- Can proof or disproof the problem.

Such good like mathematical powers possible? One better understand first that what are the limits of the quantum computers and what problems they could actually solve.

6.1 Limits of Quantum Computers

Richard Feynman was the first to propose the idea of quantum computing. The computer experts have made huge development in finding that what kind of problems, unsurpassed could be solved by the quantum computers. As per to the present day understanding, they would provide intense speedups for a limited problems like breaking the cryptographic codes on the internet that are extensively used for monetary transactions. However for some more problems such as playing chess, planning airline flights and showing theorems now strongly proposes that quantum computers as like today's standard computers would experience many of the similar limits of the algorithm.

6.2 Quantum Computers Place on the Complexity Map:

The quantum computers solves the class of problems (BQP) might relay to other classes of complexity. The BQP means "bounded-error, quantum, polynomial time" which is the class of problems of other classes namely P and NP. Examples of BQP can include the problem of factoring and the discrete logarithm. The quantum computers requires more than polynomial number of operations to solve the BQP problems as NP and NP-Complete problems supposed to lie out the BQP class. BQP might move outwards towards NP as the quantum computers being so fast can solve the problem even before a classical computer could check the answer. Computer experts recognize that BQP cannot move outside the class which is known as PSPACE that also encompasses the NP problems. PSPACE is the class of problems which can solve by taking exponential number of steps but in polynomial amount of memory. If we say that NP-Complete problems have an efficient algorithm for solving problems in this class then it means that the claim that P=NP would be correct and the representation of classes as P, NP and NP-Complete was wrong. In other words, it would mean that all the NP problems were actually P problems, making the class P equal to NP.

If we grant that P ≠ NP, than for solving NP-complete problems only hopefulness remains for solving problems in polynomial time. At first sight, quantum mechanics would seem to offer just the type of resources required. It would become possible by Quantum mechanics that vast information in stored relatively in small states of the particles called qubits.

Thus the question that "does there exists some efficient algorithm for solving NP-Complete problems" remain unsolved. Regardless of much trying, no such algorithm has been established and the computer experts have not been able to prove that such algorithm not exists. Apart from that, we can't even prove that there exists some polynomial time algorithm for solving NP-Complete problems.

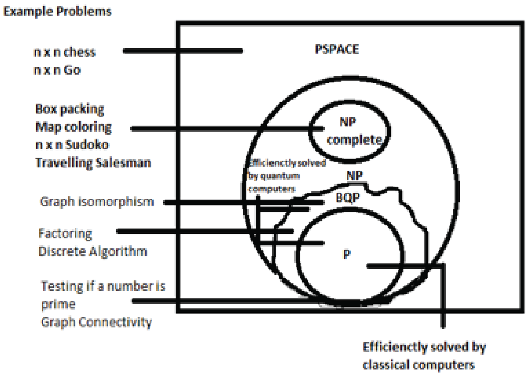

Figure 1: Complexity classes map

By considering the problems as structureless “black boxes,” exponential speedup cannot be achieved that comprises of solutions of an exponential integer which need to be verified in parallel.

Now you can assume about all the thinkable solutions in terms of quantum or more particularly in superposition. An algorithm was built by Lov Grover of Bell Laboratories in 1996 for finding the correct solution which reduced from S/2 to √2 instead √2, which was merely a speedup. This means that if you have millions of possible solutions around then instead of taking 400,000 steps you only need to take thousand steps. Nevertheless, the exponential time was not changed by taking a square root into in order to make it possible in polynomial time.

Scientists have presented that analogous uncertain speedups are the boundary for numerous other problems in addition like performing search in a list, such as totaling the number of votes in the elections and playing games such as Go or chess and the big tricks like the collision problem.

Has there been a quantum algorithm which works fast to resolve such like issues, many of the elementary structure of safe electronic trade would be of no use in a quantum computers world.

As an example performing an item search in a list is like looking for a needle in a haystack, while examining for a collision is like viewing for two indistinguishable pieces of hay, which basically exploits the issues like that, a quantum computer would actually solve.

Surely, it cannot be ruled out by the black box boundaries that the chances of an efficient quantum algorithm for NP-complete or NP-hard issues are for the future to be revealed.

Keeping into consideration the above-mentioned discussion, the conjecture that P ≠ NP not only considered by the computer experts but also that the NP-Complete problems cannot be solved in polynomial number of steps by the quantum computers.

7. Blend of Complexity with Space and Time

What can computational complexity advises us regarding the space and time? The foremost response could be “not much”: in any case, the definitions of standard complexity classes such as P and NP can be displayed as indifferent to such facts as the number of 3-D measurements moreover is the velocity of light is predictable or unpredictable. Instead, complexity theory gives idea regarding the dissimilarity amongst space and time.

The illustration of PSPACE problems includes replicating dynamical systems, determining does a regular grammar produces all possible strings, and performing an ideal strategy in two-player games such as Hex, Connect Four and Reversi. It is not difficult to display that PSPACE is minimally powerful as NP:

$$P \subseteq NP \subseteq PSPACE \subseteq EXP$$

The "EXP" above represents the class of problems that can be solved in terms of some exponential function of space and time.

The laws of physics allow us to travel back in the past. Under these considerations, we could say that just like space is a reusable resource, we can use time in the same manner so that everything in PSAPCE would come under our control.

7.1 Closed Timelike Curve:

Daniel S. Abrams and Seth Lloy in 1998 proved that NP-Complete problems can be solved efficiently by quantum computers if a small non-linear term is added to the quantum mechanics equations.

If we cannot cut the time in portions, then possibly another way to resolve NP-complete problems is to misuse time travel for efficient solution. Closed times like curves compel the physicists to work on them rather than time machines.

In crux, a CTC is a path through space and time, that means that energy could travel to happen to meet with itself in the past, creating a loop that is closed.

By what means one could practice a CTC to increase a calculation? Which means to program your computer to take nevertheless extensive time it desires to resolve the problem and then direct the answer back in time to yourself at a point before the computer started. In 1991 David Deutsch of the University of Oxford defined a model of computation with CTCs that shuns the above issue. In Deutsch's model, nature will ensure that as events unfold along the round timeline that creates the closed timelike curve, no impossibilities ever arise, a fact that can be exploited to computer program that loops.

Basically, a closed timelike curve use time and space as interchangeable computational assets. Truly, the closed timelike curve solves efficiently NP-problems and PSPACE.

For illustration, we need not to assume that the quantum computers as having supernatural powers. Nevertheless, those similar bounds can provide an extra optimistic turn. It implies that in a world of quantum computers other than the codes which could be broken by cryptographic code, persist as protected.

8. Dissatisfactions of Complexity Theory

Regardless of its descriptive scope, complexity theory has been evaluated many times. Few of the main disapprovals are mentioned below:

- Asymptotic statements becomes the basis of complexity theory but as per the facts, asymptotic statements principally not effects any kind of fixed amount can confirm or refute any asymptotic privilege.

- The conjecture P≠NP has not yet been proved and will remain as such for times to come.

- DTM becomes the base of the complexity theory, but it fails to think on other disorganized computational wonder.

- Complexity theory works on algorithm's worst-case performance and even don't tell whether it is just descriptive or contains the extreme values. For instance, may if such possibilities were picture given then maybe some NP-Completeness problems could have been identified even if P≠NP.

- The point is that, if any thinkable fault of a complexity exploration remains unanswered, either it can be a deduction that can be falsified or negated statement.

9. Future of Complexity Theory:

In order to understand the relationship of the complexity theory and its connection with the real world, we should not step back from discussing the criticisms, instead, that would be of great importance. The below-mentioned queries would advantage to all, from vigilant rational analysis:

- In what way we can describe the experimental truths on grounds of which complexity theory faiths. Barely we realize n10000 or 1.00001n algorithms, or that the computational issues humans pay attention incline to practice themselves obsessed by a relatively, number of small correspondence classes?

- In what manner the humans succeeded in huge mathematical development if P ≠NP, is there some structure which exists that can make to solve these problems easily? As theorem proving aspect is also difficult. So, if that sort of structure exits than what is that?

10. Conclusion

Could we assume that computational complexity is beneficial to philosophy or otherwise? Generalizing from the illustrations discussed above, I guess that computational complexity have a tendency to be useful when we desire to identify whether an exact fact "determines" another fact, and is not bothered about the size of the inferential sequence.

The exponential speedup of the quantum computers can be utilized but the limitation it involves along keep us optimistic as well.

New variance in the field of quantum mechanics, the quantum robots as quantum computers is the future in the field for environment interaction and their uses in our society [9].

The reason of this paper was to highlight that how philosophy could be boosted by grasping computational complexity theory into account, great as it was developed almost a period of hundred years by captivating computability theory into account.

References

[1] Great Internet Mersenne Prime Search GIMPS, https://www.mersenne.org/.

[2] How Computational Complexity Will Revolutionize Philosophy (MIT Technology Review), https://www.technologyreview.com/s/424974/how-computational-complexity-will-revolutionize-philosophy/.

[3] Aaronson, S. (2013). Why philosophers should care about computational complexity. Computability: Turing, Gödel, Church, and Beyond, 261-328.

[4] Gödel's Incompleteness Theorems (Stanford Encyclopaedia of Philosophy), http://plato.stanford.edu/bentries/goedel-incompleteness/.

[5] The Limits of Quantum by Scott Aaronson, Scientific American article, March 2008.

[6] Valiant, Leslie G." Evolvability " *Journal of the ACM (JACM)* 56.1 (2009): 3

[7] *Complexity classes, Complexity Zoo website by Scott Aaronson,* www.complexityzoo.com.

[8] *Shadows of the Mind: A Search for the Missing Science of Consciousness, Published August 22nd, 1996 by Oxford University Press, US.*

[9] Benioff, Paul. "Quantum robots and quantum computers." arXiv preprint quant-ph/9706012 (1997).

[10] Gödel’s Lost Letter and P = NP, https://rjlipton.wordpress.com/the-gdel-letter/

[11] Gruska, J. (1999). Quantum computing (Vol. 2005). London: McGraw-Hill.

[12] Nielsen, M. A., & Chuang, I. L. (2010). Quantum computation and quantum information. Cambridge university press.

[13] Mermin, N. D. (2007). Quantum computer science: an introduction. Cambridge University Press

[14] Aaronson, S. (2013). Quantum computing since Democritus. Cambridge University Press.

[15] Eisert, J., & Wolf, M. M. (2006). Quantum computing. In Handbook of nature-inspired and innovative computing (pp. 253-286). Springer, Boston, MA.