Issues with Quantum Computing

Prepared for

Professor Jason W. Ellis
New York City College of Technology

Prepared by

Daniel Romanowski
Enmanuel Arias
Lia Barbu
Nakeita Clarke
Jinquan Tan

December 21, 2020

Abstraction:

Our group has chosen issues with Quantum Computing (QC) for the Analytical Research Report. QC is a new technology which is still in the Research & Development stage. and has many issues that have yet to be solved. Qubit observation, lack of programming languages, and cybersecurity in classical computers interacting with quantum computers are some of these issues. Development cost not delivering a return on investment is also an issue, with QC having a long way to go before hitting the mainstream market.

Objective of this Paper:

This report aims to investigate and explore the current issues relating to QC as well as solutions for those issues. Through the use of qualitative methods for research, this paper will first define a quantum computer and its short-comings, and then describe quantum mechanical features. This paper will also explain the history and current state of quantum programming languages, explore the cost of a quantum computer, and analyze concerns relating to quantum cybersecurity. This collection of information will lead to a discussion, conclusion, and recommendation based on research findings.

Methods Used to Research Quantum Computing:

Methods used to research issues with Quantum Computing for this report include the using publicly available search engines. These search engines aided in the use of the reputable science and technology industry publication sites, as well as technology-based and scientific-based-blogs. Other sources include the use of the online resources offered by Ursula C. Schwerin Library, such as academic databases, academic journals, articles, and ebooks. The combined use of these resources allowed for a deep information dive that enabled us to comprehend expert insight and develop our own opinions surrounding the chosen topic for this report.

What is a Quantum Computer?

A quantum computer composes of software that processes quantum information such as quantum algorithms and quantum coding, and uses special hardware like quantum transistors, and quantum memory chips. Quantum computers are, "...physical devices that can perform high-speed mathematical and logical operations and that can store and process quantum information following the laws of quantum mechanics. When a device's processing and calculation are based on quantum information and run with quantum algorithms, it is a quantum computer. The concept of the quantum computer stems from research on reversible computers. The purpose is to address the issues regarding energy consumption of computers" (Shi, 2015, p.210). A quantum transistor's energy is passed through an electronic device at a high-speed

movement to break through the physical limits. The switch of the quantum transistor and the transistor speed control switch, has a lot of chip operation ability (more than normal), and the use of environment adaptability that is very strong. A quantum transistor is an indispensable part of a quantum computer. Quantum memory is a kind of memory with high information storage efficiency. It can assign values to any computation information in a very short time. It is an indispensable part of quantum computers and one of the most important parts of quantum computers. The effector of a quantum computer is a large control system, which can control the operation of the components. These components occupy the main position in the development of quantum computer, play an important role in the application. Quantum computers have a powerful quantum information processing capability. For the current changeable information, it can extract effective information and process it to make it become new and useful information.

Quantum information processing requires the storage and processing of the quantum computer, and then a quantum analysis of the can be given information. Using this method can accurately predict the weather conditions. At present, the computer can predict the weather conditions with 75% accuracy. However, using a quantum computer to predict the weather conditions can further improve the accuracy and make people's travel more convenient. According to Martin Gales of MIT Technology Review, ”A quantum computer harnesses some of the almost-mystical phenomena of quantum mechanics to deliver huge leaps forward in processing power. Quantum machines promise to outstrip even the most capable of today’s—and tomorrow’s—supercomputers” (Gales, 2019). At present, computers are usually attacked by viruses, which can directly cause computer paralysis and also lead to personal information being stolen. However, a quantum computer does not exist due to its non-cloned quantum principle, so users can safely surf the Internet when using a quantum computer without fear of personal information disclosure. On the other hand, a quantum computer has powerful computing power and can analyze a large number of different data at the same time, so it can accurately analyze the financial trend and play a great role in avoiding the financial crisis. It can also play a great role in the biochemistry research, can simulate the composition of new drugs, more accurate development of drugs and chemicals, so as to ensure the cost of drugs and drug potency. QC will also one day achieve Quantum Supremacy, which is explained by Gales as, “It’s the point at which a quantum computer can complete a mathematical calculation that is demonstrably beyond the reach of even the most powerful supercomputer” (Gales, 2019).

Research Results:

Our research has shown that QC has many issues that still need to be solved, or may never come to a particular resolution and will have to be adopted as standard operating procedures. Some of these issues include the way a qubit is observed, a lack of programming languages and the challenges of developing them, QC posing a threat to classical computers' cybersecurity, and development cost of QC itself. Qubit observation can be a problem due to quantum entanglement, which is also one of the main advantages of QC. Programming languages have been a challenge due to the nature of the qubit, and the way data structures behave in the quantum environment. QC poses a significant threat to classical computers' cybersecurity due to the fact that quantum algorithms can outperform classical binary discrete mathematics. Development costs are an issue because initial funding will not see a return on investment for years to come. The research to support these findings is found in the following sections.

Qubit Observation

Some of the results our research has shown is that there are issues with Qubit Observation (QO). QO involves quantum entanglement as explained by quantum theory. Written in, "Quantum Computing: A Gentle Introduction" by E. G. Rieffel and W. H. Polak, it is stated that qubits are, "... the fundamental units of information in quantum information processing in much the same way that bits are the fundamental units of information for classical processing" (Rieffel & Polak, 2011, p. 9). To be brief, quantum theory states that a qubit can be entangled with another qubit, and with a complete lack of physical connection. When one of these qubits goes through a state of change, the other entangled qubit's state will change the same as the first. Meaning that they are one in the same, in layman's terms.

When it comes to QO, Rieffel & Polak explain, "This behavior of measurement is an axiom of quantum mechanics. It is not derivable from other physical principles; rather, it is derived from the empirical observation of experiments with measuring devices. If quantum mechanics is correct, all devices that measure single qubits must behave in this way; all have associated bases, and the measurement outcome is always one of the two basis vectors" (Rieffel & Polak, 2011, p. 17). Since QO is established by observation of experiments, and not by principles known, the two vectors mentioned in the latter, can easily be affected by temperature change when under observation. The reason for this being that if we have two entangled qubits, and one goes through a temperature change, then this will affect the other. To avoid this, quantum chips operate at extremely cold temperatures, which is addressed in CNBC's video, "... the temperature of the quantum chips themselves, they need to be kept at temperatures colder than interstellar space, closer to absolute zero" ("The Hype Over Quantum Computers, Explained," 2020).

Quantum Programming Languages

QC is gaining traction again due to recent developments on both the hardware and software front. Programming languages for QC are referred to as Quantum Programming Languages (QPLs). The earliest notation of QPL was in 1936 and the work of Garrett Birkhoff and John von Neuman who were focused on quantum logic. After that came the invention of linear logic by Jean-Yves Girard in 1987. Nine years later, the first practical QPL, quantum pseudocode, was proposed by Emanuel Knill. In that same year, quantum lambda calculus was the language put forward by Philip Maymin. Yet Quantum Computation Language (QCL), created by Bernhard Ömer in 2003 is recognized as the first real tangible QPL. Andre van Tonder developed a quantum lambda calculus as a QPL option and Peter Selinger presented Quantum Flow Charts (QFC).

Even though there has been some semblance of QPL evolving, there are important fundamental issues that need to be addressed in order for QC to transcend its current programming limitations. Key issues facing the development of QPLs include the fact that "quantum mechanics itself (and by extension quantum information theory) is incomplete. Specifically missing is a theory of measurement" (Donald, 2008). This issue is akin to building a house without a completed blueprint, providing great margin for lots of errors, unforeseen issues, the potential of loss of time and money. "A second key source of difficulty is the lack of quantum computing hardware for running quantum algorithms" (Donald, 2008).

While several technology companies are trying to get a head start with quantum computing, doing so by means of using quantum computer simulators using traditional computer hardware still doesn't provide the conditions necessary for developing a QPL with accuracy. Consider how accurate a piece of software would be for a Mac laptop if it were developed from a Windows laptop running Max OS on a virtual machine. On a more linguistic approach, data types have not yet been fully realized in many of the current QPLs, much like the progression of quantum computing itself, QPLs are still in their infancy. However, there are signs growth much might be quicker than that of when computer software development was in its infancy. Given the cloud computing and virtual computing technologies available, the growth of quantum computing might surpass the limitation of progress due to lack of hardware. Some technology companies focused on quantum computing are considering a path similar to IBM's mainframe rental service and the more modern version Amazon Web Service, where the company builds a farm of quantum computers then sells on-demand access to the quantum computer's hardware resources.

Another noticeable limitation of QPL is that, “The quantum computing stack looks more like Swiss cheese. The community has succeeded in designing the top (programming language) and the bottom (qubits and gates) layers but the middle of the stack contains countless holes” (Brierley, 2019). ‘The middle of the stack’ as Brierley refers to it, is the equivalent of assembly language for computer programming. It is possible that once the issues surrounding qubits have been resolved, then there may be an emergence of quantum assembly languages. It would be interesting to see what languages emerge that caters to both the machine language for quantum computers as well as the classic computer. This isn’t an indication that current high-level QPLs are close to perfect. “Another important design issue for QC programming languages is the language’s approach to data typing. ‘Data typing’ refers to programming language constructs that label the kind (or type) of data that a program or function expects, and allows the function to use the type of the data to determine how to perform a specific operation” (*National Academies of Sciences, Engineering, and Medicine,* 2019). High-level languages allow programmers to write instructions using English-like statements. They also allow for the same program to run on more than one computer. Classical programming languages have suitable data types such as integers, floating point numbers and characters, while these data types are capable for use in QPLs, there aren’t any attributed directly to processing quantum logic (Dirac notation, matrices, gates, operators, etc.).

As of June 2020, there has been news of two new QPLs signifying progress closer to more and better programming for quantum computers. Silq, created by computer scientists at ETH Zurich, and QUA, created by Quantum Machines, have made claims that could mean they have broken through a pragmatic barrier for QPLs. In the article, ”The first intuitive programming language for quantum computers,” by Florian Meyer, Martin Vechev, one of the creators of Silq, is quoted saying that Silq "...allows programmers to utilize the potential of quantum computers better than with existing languages, because the code is more compact, faster, more intuitive and easier to understand for programmers" (Meyer, 2020). While QUA has been described as "a standard universal language for Quantum Computing,” (“Quantum Machines Announces QUA As First Standard Universal Language for Quantum Computers,” 2020), it is too early to truly determine the strength of either claims.

Development Cost

QC is the newest and promising technology on the market. It comes with the significant potential to overcome classical computers in solving complicated operations. One of the challenges that QC has to overcome to become an asset for everybody is the development cost. The main costly thing is that quantum structures are unstable. SaBine Hossenfeld states in her article, “ Quantum supremacy is coming. It won’t change the world” that, “The trouble is that quantum systems are exceptionally fragile. To maintain their quantum behavior, qubit chips are enclosed in sealed boxes fitted with vacuum pumps to remove stray air molecules, or cooled to a fraction of a degree above absolute zero. More qubits means more fridges, more connections and more expense” (Hossenfeld, 2019). Quantum computers to be able to execute something useful

need almost a few million qubits. The quantum computer that achieved quantum supremacy, Google's Sycamore, has a 53 qubits processor.

Hossenfelds explains in the same article, "Today, a single qubit will set you back $10,000 – and that's before you consider research and development costs. At that price, a useful universal quantum computer – hardware alone – comes in at at least $10bn. This for a machine whose true commercial value is far from guaranteed. To make quantum computers commercially viable, the cost per qubit will have to drop dramatically" (Hossenfeld, 2019). It seems like it is almost impossible for quantum computers to enter the mainstream market. Researchers look for ways to solve this problem and decrease the development cost to make quantum computers reasonable and ready for mass production. One of the solutions that the researchers found is to make quantum computers workable at a higher temperature. In their research report, a group of scientists from Australia state that, "...we have presented a fully operable two-qubit system in an isolated quantum processor unit cell that allows operation up to 1.5 K—a temperature that is conveniently achieved using pumped 4He cryostats—in which we reach near fault-tolerant single-qubit fidelities" (Yang et al., 2020). Being able to keep the temperature at 1.5 K will help to reduce the cost of refrigeration. Millions of dollars are spent maintaining the temperature to 0.1 K. The 1.5 K temperature can be maintained with a few thousand dollars. It is a significant accomplishment; it represents a step forward in making quantum computers a reality.

Quantum Computers and Cybersecurity

Setting aside the developmental issues plaguing quantum computers, one of the other problems the tech industry faces within the quantum realm is cybersecurity. As Shi (2015) stated, quantum computers are capable of performing high-speed mathematical and logical operations. Imagine if a malicious individual used quantum computing technology to crack the encryption used to keep passwords and sensitive information unreadable from prying eyes. Assuming a system uses a 128-AES (Advanced Encryption Standard) cipher, it would take about 10.79 quintillion years to crack the cipher if you could amass the computing power to test a trillion keys per second (Wood, 2011). According to Wood, quantum computing technology under the same assumptions could crack the cipher within six months. We could see a quantum computer within the next 20 years at the current development rate, so enterprises need to start taking action to future proof their systems.

According to Mone (2020), "...there is always a risk cryptography can be broken" (p. 13), so it is essential to be continuously developing new and innovative algorithms to ensure that our systems can be properly encrypted and protected for years to come. So NIST (National Institute of Standards and Technology) called on its members and the cryptography community to test several proposed algorithms. By 2019, out of the initial 89 proposals, 29 candidates remained. One of the proposed solutions to this problem is through the use of lattices, an arrangement of points connected to each other (Mathematics, 2020), to find the point closest to the lattice design's origin. "In two dimensions, the lattice problem is relatively simple to solve…" (Mone, 2020), but it can quickly become a complicated problem to solve when you drastically increase

the dimensional planes. This approach is highly regarded among the cryptography community because researchers have been testing and attacking it for many years without successfully cracking it.

In the end, there will likely be more than one approach selected by NIST because there is not a single perfect way to resolve the dilemma. Also, there are many preferences to consider when developing a new encryption standard in a post-quantum world. NIST is also evaluating other aspects of the candidates, such as their implementation in the real world, resource management, speed, and key size. Post-quantum cryptography aims to develop systems protected against quantum and classical computers (Chen, et al., 2016). These new systems will also incorporate existing communication protocols and networks. Post-quantum cryptography systems will require additional resources that smartphones or servers may be able to handle. Still, embedded systems may have difficulty providing enough resources due to their innate limitations (Mone, 2020). More importantly, NIST has advised the transition from "key sizes and algorithms that provide 80 bits of security…" (Chen, et al., 2016) to those that provide up to 128 bits of security. However, NIST expects this transition to only hold up until 2030, when researchers have estimated the first quantum computer capable of cracking 2000 bit RSA (Rivest-Shamir-Adleman) within a few hours will be built. According to Chen et al., once these new encryption standards have been developed, NIST will reevaluate the current encryption standards and determine whether to deprecate or withdraw their initial remarks. So enterprises must continue their efforts to update and ensure plans are in place to future proof their systems once post-quantum standards have been established.

Discussion of Research Results:

It is said that necessity is the mother of invention. What is the necessity of QC? We need QC in order to advance our knowledge of the universe around us. Modern computers are reaching their limit of computational power, and new types of hardware and software are needed to meet the challenges of the future's processing demands. Through there are physical and monetary limits to today's development of QC, this technology is the next step in computer evolution. The following discussion is made to clarify our group's research in why QC is an important pursuit.

A discussion to be made about QO is that in order to operate, quantum computers need to be kept close to absolute zero. Not only is this a problem for a large corporation with plenty of resources to address, but this also affects who will be able to own, and operate a quantum computer when they are ready for the general market. Hossenfeld makes this point in her article cited above, when she wrote, " Today, single qubit will set you back $10,000... " (Hossenfled, 2019). But Hossenfeld also uses the word, "Today..." (2019), and this is a very important part of this discussion. Today's quantum computers are very expensive to build and operate, but that says nothing about the future of QC costs. If one could imagine the resources and costs that would have been involved for the computers of the 1950's to meet today's processing functions, the price would have been astronomical, while using all of the resources available at the time.

Another discussion to be made is about QPL. At the moment all of the QPLs are run on separate frameworks, they are not universal, and each programming language is specific to the quantum computer it is being developed for. The QPLs available either focus on function (handle symbolic computation, mathematics-based functions) or procedure (process in order of a set of command). This means that the current offering of QPLs has a basis in classical computer languages and have been more adept to classical computer science. This means that current QPLs have yet to build an extension which successfully caters to the physics of QC.

Though these daunting issues are present, STEM is no stranger to addressing and also solving problems. Some researchers have already begun offering their suggestions for solving some portion of the shortage of sound QPLs.

"A QPL should or should possibly

i. run on top of a simulator as well as on a real system,
ii. help in discovering new efficient quantum algorithms,
iii. enable a layperson to write quantum programs, comply with the concept of abstract data types (ADTs),
iv. provide high-level language constructs,
v. support quantum data and quantum control,
vi. support programming in the large and programming communication processes,
vii. be as close as possible to classical language concepts for pragmatic reasons and
viii. support quantum processes completely, including measurement"

(Roland, 2007).

Conclusions:

A conclusion drawn from this research is that, like other developing technologies, QC has its advantages and disadvantages. The advantages of QC is that it will have greater computational power, faster processing time, and the ability to use quantum mechanics properties. Disadvantages could be seen as development cost, and the mysteries surrounding the quantum environment which require more research. Although some of these tasks may seem daunting, with time, any issues surrounding QC development may be solved.

There is also a need for standardized operating systems, but high-level QPLs have to be developed first. Once high-level QPLs become readily available, then companies will be able to use these languages to create operating systems to develop applications that use quantum algorithms. These applications could be similar to ones found on Windows, and Mac, such as Microsoft Access, or Keynote. Also there could possibly be applications that run on multiple operating systems like Adobe, and Mozilla Firefox.

Recommendations:

Our group recommends that companies stay the course with current QC development, and challenges. QC is just at the beginning of its evolution, and it will be years before any useful applications are made with it. It can be argued that current development of QC is much like the development time-line we saw with classical computers. With classical computers starting out with large machines that did not have that much processing power, and taking 20-30 years before becoming more of a useful tool to main-stream society. QC may never be something that an individual has possession of themselves, and may only be something people use classical computers to log into. However, it is evident that as long as development continues at its current pace, QC will be a common practice in the near future and could usher in a new age of human existence much like the industrial revolution.

References:

Brierley, S. (2020, January 2). *The conundrum of a quantum computer.* Electronics Weekly. https://www.electronicsweekly.com/news/the-conundrum-of-a-quantum-computer-2020-01/

Chen, L., Smith-Tone, D., Peralta, R., Moody, D. et al. (2016, April). *Report on Post-Quantum Cryptography.* National Institute of Standards and Technology. http://doi.org/10.6028/NIST.IR.8105

CNBC. (2020, Jan 10). *The Hype Over Quantum Computers, Explained.* [Video]. YouTube. https://youtu.be/u1XXjWr5frE

Dale, N., & Lewis A.J. (2016). *Computer Science Illuminated* (6th edition). Jones & Bartlett Learning.

Gales, Martin. (2019, Jan. 29). *Explained; What is a quantum computer?* MIT Technology Review. https://www.technologyreview.com/2019/01/29/66141/what-is-quantum-computing/

Hossenfelds, S. (2019, August 2). *If quantum computers are to help solve humanity's problems, they will have to improve drastically.* The Guardian. https://www.theguardian.com/technology/2019/aug/02/quantum-supremacy-computers

Lardinois. F. (2020, June 15). *Silq is a new high-level programming language for quantum computers.* TechCrunch. https://techcrunch.com/2020/06/15/silq-is-a-new-high-level-programming-language-for-quantum-computers/

Mathematics. (2020). *Encyclopædia Britannica.* https://academic-eb-com.cilevels/collegiate/article/mathematics/109827#335981.toc

Meyer, F. (2020, June 15). *The first intuitive programming language for quantum computers.* DevTopics. https://www.sciencedaily.com/releases/2020/06/200615115820.htm

National Academies of Sciences, Engineering, and Medicine. (2019). *Quantum computing: Progress and prospects.* The National Academies Press. https://doi.org/10.17226/25196.

Quantum Machines. (2020, June 18). *Quantum Machines Announces QUA As First Standard Universal Language for Quantum Computers.* Cision PR Newswire. https://www.prnewswire.com/il/news-releases/quantum-machines-announces-qua-as-first-standard-universal-language-for-quantum-computers-301079334.html

Rieffel, E. G., & Polak, W. H. (2011). *Quantum Computing : A Gentle Introduction*. MIT Press. https://ebookcentral.proquest.com

Roland, R. (2007). Quantum programming languages: An introductory overview. *Computer Journal, 50*(2), 134–150. https://doi.org/10.1093/comjnl/bxl057

Shi, D., Guo, Z. & Bedford, N. (2015). *Micro and Nano Technologies.* William Andrew Publishing. https://www.sciencedirect.com/topics/engineering/quantum-computer

Sofge, D. A. (2008). *A Survey of Quantum Programming Languages: History, Methods, and Tools*. Second International Conference on Quantum, Nano and Micro Technologies (ICQNM 2008). doi: 10.1109/ICQNM.2008.15

Toady, T. (2010, June 29). QCL: *Obscure programming language of the months.* ScienceDaily. http://www.devtopics.com/qcl-obscure-programming-language-of-the-month/

Valiron, B., Ross, N. J., Selinger, P., Alexander, D. S., & Smith, J. M. (2015). Programming the quantum future. *Communications of the ACM, 58*(8), 52–61. https://doi.org/10.1145/2699415

Wood, L. (2011, March 21). *The Clock Is Ticking for Encryption.* Computerworld. https://www.computerworld.com/article/2550008/the-clock-is-ticking-for-encryption.html

Yang, C. H., Leon, R. C. C., Hwang, J. C. C., Tan, K. Y. et al. (2020). Operation of a silicon quantum processor unit cell above one kelvin. *Nature*, *580*(7803), 350–354. https://doi.org/10.1038/s41586-020-2171-6