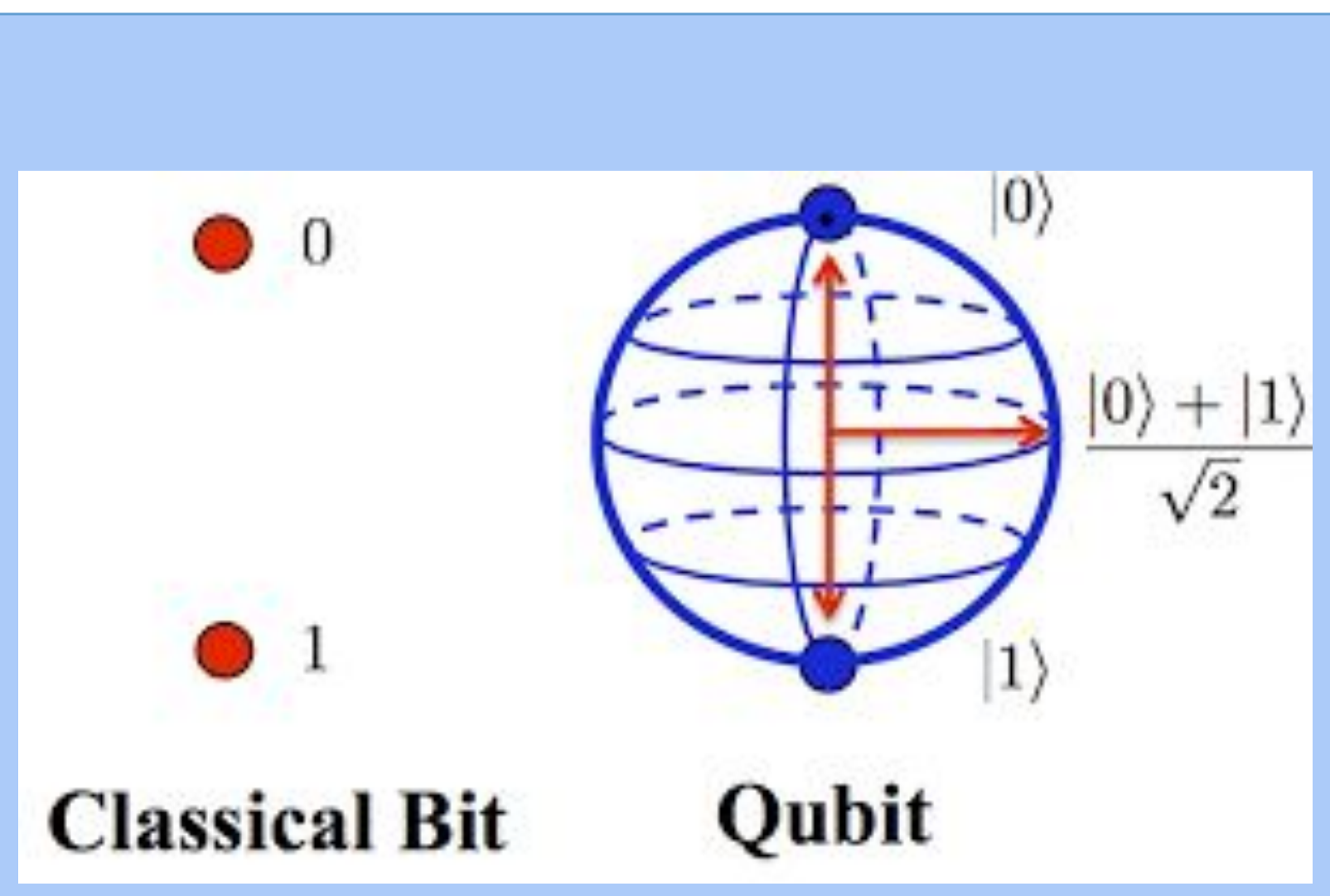
Impact of Quantum Computing on Present Day Cryptography

Arifulla Shaik, Irvine Valley College
Mentor: Lan Pham

Abstract

Current computers are lightning-fast for most applications; however, they are no match for quantum computers that are being developed. While a computer's processor can only handle one task at a time, a quantum computer is capable of computing multiple tasks simultaneously due to its properties of quantum physics. Google, in partnership with NASA, has successfully demonstrated that a quantum computer can compute a calculation in seconds what would take even the world's most powerful computers thousands of years, achieving Quantum Supremacy. However, quantum computers are a potential risk to public-key encryption (AES 256-bit), as they could compromise the security of digital communication over the internet. Modern cryptography is primarily based on the fundamental concept of prime factorizations. As of today, supercomputers are incapable of breaking encryption, as it would take them thousands of years. However, a quantum computer could potentially break encryption within a reasonable time using Shor's algorithm on a quantum computer.

Quantum Computer



Background and Introduction

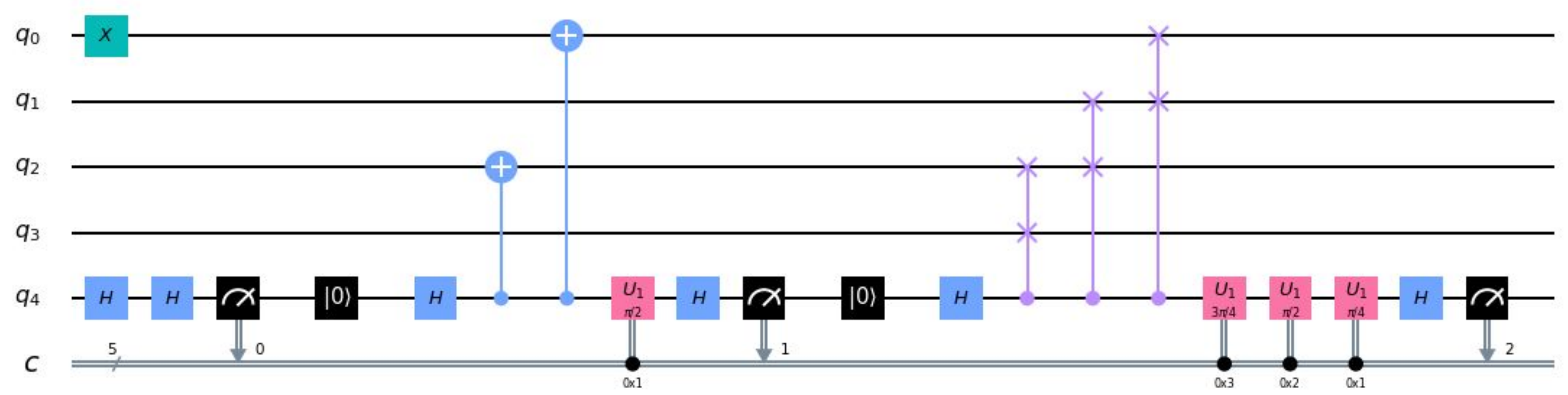
The objective of this study is to discover the implementation of quantum computing in modern cryptography and examine post-quantum algorithms. The post-quantum cryptography is a chain of cryptographic frameworks that are created as an approach to secure against an attack by both quantum and classical computers. However, as of today, this is no longer true because most mainstream public-key encryption can be effectively decoded by utilizing a powerful quantum computer. This is because modern cryptography is primarily based on the fundamental concept of prime factorizations. While it is extremely hard and time-consuming to find prime factors of a large number on a classical computer, it is much easier to find the prime factors using Shor's algorithm on a quantum computer. The most efficient classical factoring algorithm currently known is the General Number Field Sieve. This algorithm has an exponential asymptotic runtime, whereas, Shor's algorithm has a polynomial asymptotic runtime. This is an enormous difference between these two algorithms as Shor's algorithm will always find the prime factors of a number, regardless of its size, faster compared to a classical computer. As a result, with the combination of effective quantum algorithms and the growth of quantum computer, it will have a significant impact on modern-day cryptography. Our study will attempt to highlight the meaningful impact quantum computers will have on cryptography that can potentially change the landscape of modern computing forever.

Methodology

Shor's algorithm consists of two parts:

1) A reduction of the factoring problem to the problem of order-finding (Classical Part)
   a) Initialize the composite integer *n* for factoring
   b) Select sample of (random) integers *a*
   c) Use the Euclidean algorithm to find the GCD
      i) If GCD(*a*,*n*) ≠ 1, then there is a nontrivial factor of *N*
      ii) If GCD(a,n) = 1, find *r*, the period of the function, using the period-finding subroutine (f(x) = *a*x mod *n* where f(x + *r*) = f(x))
   d) The factors of *n* are GCD(*ar*/2 ± 1,*n*)
2) A quantum algorithm to solve the order-finding problem (Quantum Part)
   a) Estimate the period of *r*(*a*) by mapping the problem into QFT
   b) Convert *(a,n)* into binary, construct the quantum circuit of n + 1 qubits
      i) If r is even, calculate factor p = a^r/2 - 1.
      ii) Calculate second factor q = n/p

output of the quantum circuit

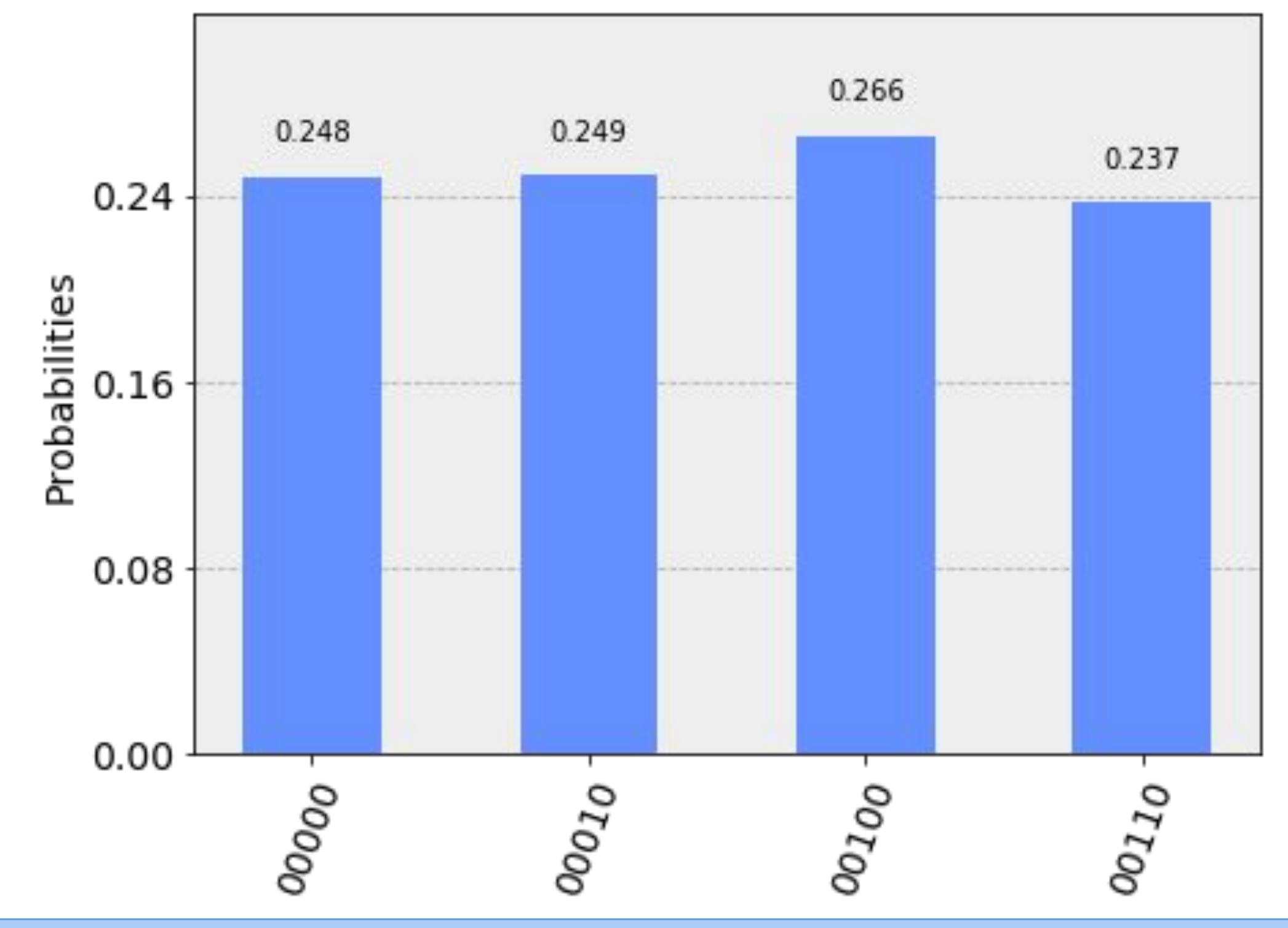


Results

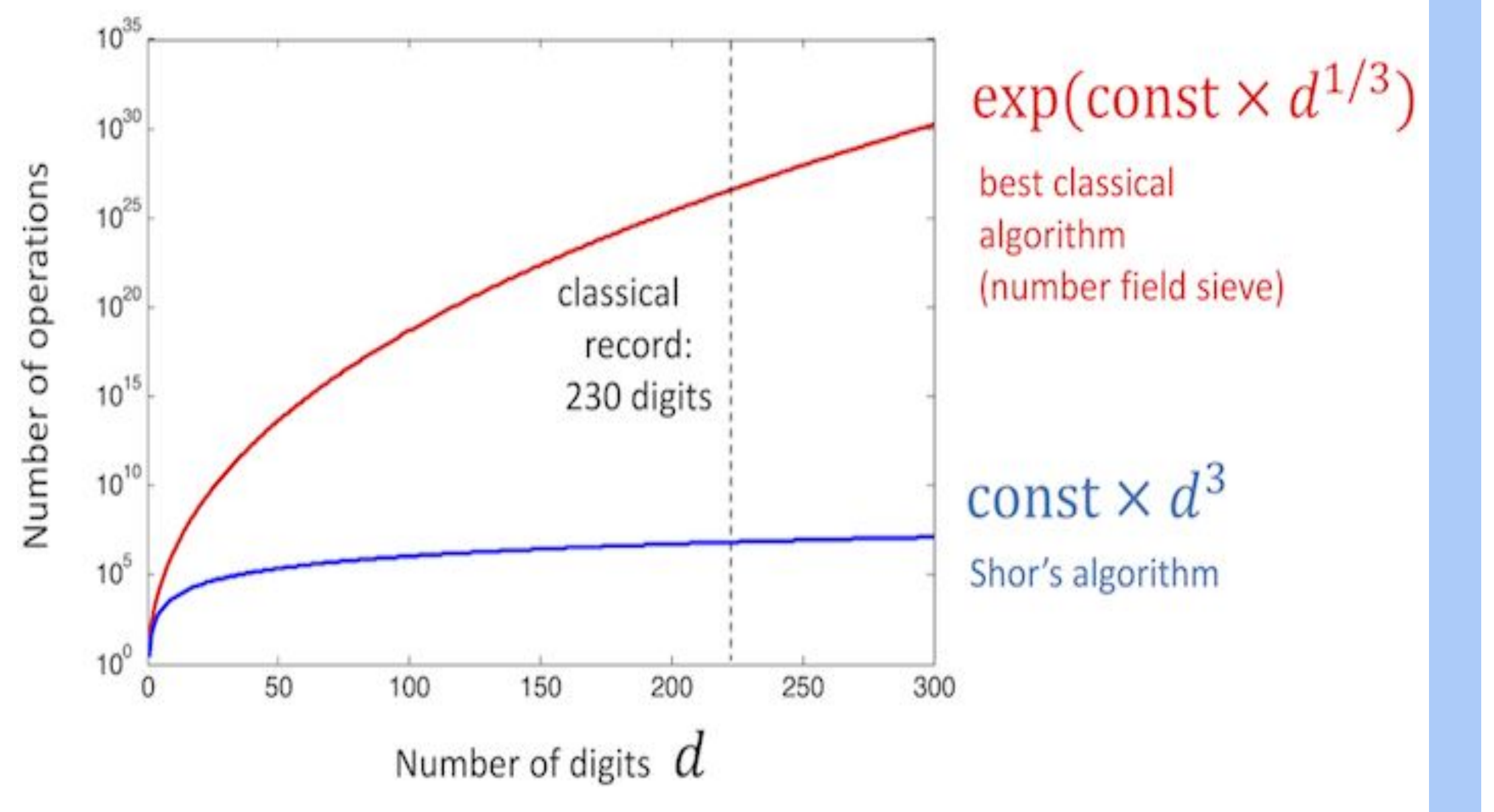
Using different value of a with n=15 to compute Shor's algorithm

a	Period r	$\gcd(15, a^{r/2} - 1)$	$\gcd(15, a^{r/2} + 1)$
1	1		
2	4	3	5
4	2	3	5
7	4	3	5
8	4	3	5
11	2	5	3
13	4	3	5
14	2	1	15

Results of 5 qubit on IBM quantum computer



Complexity of factoring



Conclusion

The results were generated through a 5 qubit hardware and simulation which can indicate that a quantum computer can potentially be able to break RSA encryption because they are successful at finding the prime factors of a large number in a reasonable time compared to normal computers. 2048-bit RSA is a standard cryptographic algorithm that helps to keep information safe and hidden on the Internet. This method is very hard to break the encryption. The basic concept of the 2048-bit RSA is that it is based on a simple idea, which is prime factorization. While the current quantum computers that are available today cannot break 2048-bit RSA since we need a lot more qubits and that might come really soon in the future, but today's quantum computers are still faster than some of the supercomputers. Shor's algorithm massively reduces the complexity of breaking RSA using a quantum computer from exponential complexity to polynomial complexity. And according to other studies, a quantum computer with 4099 perfectly stable qubits could break the RSA-2048 encryption in 10 seconds which would have taken some of the supercomputers 300 trillion years to complete. However, when we run the algorithm on an actual quantum computer, we see that most of the results point toward the qubit 00100 and there smaller number results that point to different qubits, it is not uniform due to present errors in quantum computers. This is because quantum computers are difficult to engineer, build and program. As a result, they are filled with errors in the form of noise, faults, and loss of quantum coherence. Decoherence also has a great impact on quantum error, which is caused by vibrations, temperature fluctuations, electromagnetic waves and other interactions with the outside environment, ultimately destroying the quantum state of a qubit. One way to reduce errors is through Measurement Error Mitigation. This process predicts any possible errors of the results in advance and uses that information to invert the results of an unknown computation.

References

1) Polynomial-Time Algorithms for Prime Factorization and Discrete
Logarithms on a Quantum Computer. arXiv:quant-ph/9508027

2) Monz, T., Nigg, D., Martinez, E.A., Brandl, M.F., Schindler, P., Rines,
R., Wang, S.X., Chuang, I.L. and Blatt, R., 2016. Realization of a scalable Shor algorithm. Science, 351(6277), pp.1068-1070.

3) A. Y. Kitaev, http://arxiv.org/abs/quant-ph/9511026 (1995).

4) Fast Quantum Modular Exponentiation Architecture for Shor's Factorization Algorithm. arXiv:1207.0511

5) Circuit for Shor's algorithm using 2n+3 qubits. arXiv:quantph/0205095

6) IBM Quantum Experience Documentation/Full User Guide. https://quantumexperience.ng.bluemix.net/qx/tutorial?sectionId=fulluser-guide&page=introduction

7) https://github.com/QISKit/qiskit-tutorial

8) https://github.com/QISKit/qiskit-sdk-py