Implementation and Analysis of Shor's Algorithm to Break RSA Cryptosystem Security

Siyon Singh [*] and Eric Sakk

December 12, 2023

Abstract

The ongoing development of Quantum Computing (QC) is a breakthrough paradigm that holds the power to revolutionize the way we process information and solve complex problems today. Quantum Computation works on the fundamental principles of Superposition and Entanglement, enabling Quantum Computers to Leverage the power of Physics in Computing. QC has the potential to revolutionize fields such as Drug Development, Financial Modeling, Optimization, and Cryptography. Shor's algorithm, a quantum algorithm used to factorize large numbers, poses significant threats to RSA, a widely used public key cryptosystem. RSA relies on the difficulty of factoring large semi-primes to keep its security. Shor's algorithm's ability to factorize quickly on a quantum computer undermines RSA's security assumptions, necessitating the exploration of post-quantum cryptographic solutions to ensure secure communication in the quantum era. The goal of this paper is to assess the present state of Shor's algorithm implementation and efficiency, specifically its application in the context of compromising public key cryptosystems like RSA.

Keywords— Quantum Computing, Cryptosystems, RSA, Shor's Algorithm, Quantum parallelism, Superposition, Qubits, Computational basis, Hilbert space, Quantum measurement, Born's rule, Quantum Fourier transform, Quantum error correction, Decoherence, Quantum cryptography, RSA cryptanalysis

1 Introduction

In today's digital age, ensuring secure communication and data protection is of utmost importance. Public key cryptosystems or asymmetric cryptography have played a pivotal role in achieving this goal by providing a framework for secure data transmission over unsecured channels. In public-key cryptography, a set of two keys is generated (public and private keys) to decrypt and encrypt the data. Rivest–Shamir– Adleman (RSA), the most widely used asymmetric algorithm today [1]. The strength of the RSA algorithm depends on the difficulty of factoring large integers in polynomial time [2], making it computationally exhausting to break with classical computers. However,

[*] Corresponding author: Siyon Singh, Email: siyonsingh2007@gmail.com

the development of QC has been raising concerns about the vulnerability of RSA and other classical cryptographic schemes.

The RSA encryption relies on two mathematically correlated keys, a public key, which is used for encryption, and a private key used for decryption [3]. The security of RSA is based on the difficulty of factoring the product of two large prime numbers, a task computationally exhausting for classical computers when the numbers are sufficiently large [3]. This complexity forms the foundation of RSA's security, allowing sensitive data to be safely transmitted over the internet. In 1994, an American mathematician, Peter Shor proposed a quantum algorithm for factoring large integers, which runs on a polynomial-time [3], it is a quantum algorithm that poses a significant threat to RSA and other classical cryptosystems. It uses the properties of quantum parallelism and entanglement, allowing it to efficiently find the prime factors of large numbers that RSA relies on for its security.

In a classical context, breaking RSA by factoring large numbers is a time-consuming process, especially when dealing with keys of considerable length. On the other hand, Shor's algorithm has the potential to perform this task exponentially faster by enforcing the quantum properties of superposition and interference. Consequently, RSA-encrypted messages could be deciphered quickly with the advent of practical quantum computers.

The primary objective of this research paper is to assess the current state of implementing Shor's algorithm as applied to breaking public key cryptosystems like RSA. Examining recent developments in the field of Quantum computing and cryptography, we seek to provide insights into the potential implications for RSA's security in the era of Quantum computing.

The scope of this study encompasses a comprehensive review of the progress made in implementing Shor's algorithm on existing quantum computers. We aim to analyze the challenges faced in quantum computation, such as error correction and resource requirements, and evaluate the efficiency of Shor's algorithm compared to classical factorization methods.

By achieving a comprehensive understanding of the current state of Shor's algorithm implementation, this research paper will contribute to the ongoing efforts to develop secure and quantum-resistant cryptographic solutions. The knowledge gained from this study will assist in making informed decisions regarding the future of RSA and the adoption of quantum-safe cryptographic techniques to safeguard sensitive information in the quantum era.

2 Shor's Algorithm and Quantum Computation

2.1 Quantum Parallelism and Superposition

Classical computing is limited by storage capacity and speed of calculations [4], even when parallel computation is conducted on it [4]. Quantum parallelism is a foundational concept in quantum computing, allowing quantum systems to perform multiple computations simultaneously. Unlike classical bits that are limited to states of 0 or 1, quantum bits (Qubits) can exist in a superposition of both states [1]. Quantum Parallelism allows the computation to be performed in exponentially low time as compared to the conventional method [5]. The significance of quantum parallelism lies in its potential to solve problems that are intractable for classical computers due to their exponential complexity.

Superposition is a core phenomenon in quantum mechanics that forms the basis of quantum computing [1]. It refers to the ability of Qubits to exist in a linear combination of multiple states simultaneously [3]. The power of superposition becomes evident when multiple Qubits are considered. An n-Qubit quantum computer can be in a superposition of 2^n states or computational basis [2], exponentially increasing the system's information-carrying capacity compared to classical systems.

In quantum physics, a complete description of the state of this system requires $2^n - 1$ complex numbers [6]. To be more precise, the state of the quantum system is a point in a 2^n - dimensional vector space [6]. The Hilbert space associated with this quantum system is the complex vector space with these 2^n states as basis vectors, and the state of the system at any time is represented by a unit-length vector in this Hilbert space [6]. As multiplying this state vector by a unit-length complex phase does not change any behavior of the state, we need only $2^n - 1$ complex numbers to completely describe the state. We represent this superposition of states as Equation(1):

$$\sum_{i=0}^{2^n-1} a_i \left|S_i\right\rangle \tag{1}$$

where the amplitudes a_i are complex numbers such that $\sum_i \left|a_i\right|^2$ and each $\left|S_i\right\rangle$ is a basis vector of the Hilbert space.

$$\left|\psi\right\rangle = a_0 \left|S_0\right\rangle \ + a_1 \left|S_1\right\rangle \ + a_2 \left|S_2\right\rangle \ + ... + a_{2^n-1_c} \left|S_{2^n-1_c}\right\rangle \tag{2}$$

In Equation(2), the values α_0, α_1, α_2 ... $\alpha_{2^n-1_c}$ and are the complex probability amplitudes. When we measure the Qubit, ***Born's rule***, it states that if an observable corresponding to a self-adjoint operator is measured in a system with a normalized wave function, the result will be one of the eigenvalues of that self-adjoint operator[1]. After Quantum Measurement We can say that the superposition state of Qubit (ψ) will collapse to a particular eigenstate $\left|S_i\right\rangle$ with the Born probability as shown in Equation(3)

$$\text{Probability of collapsing on Eigenstate} \left|S_i\right\rangle = \left|a_i\right|^2 \tag{3}$$

The probability amplitudes α_0, α_1, α_2 ... $\alpha_{2^n-1_c}$ also satisfies the normalization condition:

$$\left|\alpha_0\right|^2 + \left|\alpha_1\right|^2 + \left|\alpha_2\right|^2 + ... + \left|\alpha_{2^n-1_c}\right|^2 = 1 \tag{4}$$

According to Equation(4), this normalization ensures that the probability of measuring the Qubit in either of the states $\left|S_0\right\rangle$, $\left|S_1\right\rangle$, $\left|S_2\right\rangle$ ... $\left|S_{2^n-1_c}\right\rangle$ is unity.

2.2 Analysis of Shor's Algorithm and RSA

Shor's algorithm addresses the formidable challenge of factoring large composite numbers into their prime factors, a problem of immense significance in number theory and cryptography. Classically, this problem becomes increasingly time-consuming as numbers grow larger due to its exponential time complexity. Experimentally, its implementation is highly demanding because it requires both a sufficiently large quantum register and high-fidelity control. Shor's algorithm, utilizing the principles of quantum parallelism and superposition, offers a groundbreaking solution by drastically reducing the computational effort required for factorization.

2.2.1 Working of Shor's Algorithm

In the context of classical factoring, consider a value to be factorized, $N = 15$, for instance. To initiate the process, a random integer 'a' is selected from the range $[2, N-1]$. Suppose we take $a = 7$. An assessment is made to ascertain whether the greatest common divisor, $\gcd(a, N)$, equals 1[7]. If it does not, a factor is immediately identified. The potential values for 'a' in this scenario encompass $\{3, 5, 6, 9, 10, 12\}$[7]. Subsequently, modular exponentiation is performed by calculating the values of $(a^x \mod N)$ for $x = 0, 1, 2$, and so on, until the period 'r' is determined. The period 'r' denotes the smallest positive value of x for which $(a^x \mod N) = 1$[7]. The next step involves finding the factors of N using the greatest common divisors of $(a^{r/2} \pm 1)$ and N. This is efficiently achieved through classical techniques such as Euclid's algorithm.[7]

In the quantum realm, the process involves calculating $(a^x \mod N)$ within a computational register for varying x values [7]. Subsequently, the period 'r' is extracted using the quantum Fourier transform (QFT) applied to the period register [7]. Remarkably, the QFT allows for the extraction of the period from a limited number of measurements, which does not escalate in proportion to the magnitude of the number being factorized.

2.2.2 Working of RSA Algorithm

The RSA algorithm is based on mathematical concepts such as modular arithmetic, prime factorization, and exponentiation. It relies on the computational complexity of factoring large numbers into their prime factors, a task considered difficult for sufficiently large numbers.

Key Generation

The key generation process involves selecting two distinct prime numbers, p and q, and computing their product $N = p \times q$. The totient function $\phi(N)$ is then derived as $(p-1) \times (q-1)$, representing the count of numbers less than N that are coprime to N.

Next, the algorithm selects an encryption key e such that e is coprime to $\phi(N)$ and $1 < e < \phi(N)$. This pair (e, N) forms the public key, used for encryption. The private key d is computed as the modular multiplicative inverse of e modulo $\phi(N)$, i.e., $d \times e \equiv 1 \pmod{\phi(N)}$. The private key (d, N) is kept secret and used for decryption.

Encryption and Decryption

Encryption involves raising a plaintext message M to the power of e modulo N, resulting in the ciphertext $C \equiv M^e \pmod{N}$. Decryption utilizes the private exponent d to raise the ciphertext C to the power of d modulo N, retrieving the original plaintext: $M \equiv C^d \pmod{N}$. The security of RSA relies on the difficulty of factoring the modulus N into its prime factors, which forms the basis for its strength as an asymmetric encryption scheme.

2.2.3 Quantum Fourier Transform (QFT)

Let us first review the classical discrete Fourier transform. The Classical discrete Fourier transform acts on a vector $(x_0, x_1, x_2, \ldots, x_{n-1})$, mapping it to another vector $(y_0, y_1, y_2, \ldots, y_{n-1})$, when both vectors are elements of the complex space [1]. The quantum Fourier transform is a linear transformation on quantum bits. It is analogous to the classical inverse discrete transform [1]. The quantum Fourier transform operates on some quantum state $|x\rangle$ and maps it to a quantum state $|y\rangle$ using the formula:

$$y_k = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x_n \omega_N^{nk} \tag{5}$$

In Equation(5), the n value will increment from 0 to $N-1$. The is a rotation, and N is the length of vectors where $N = 2^n$[1]. The quantum Fourier transform is important for many reasons, not only is it related to the Hadamard gate (A Quantum Gate that puts the Qubit in an equal superposition state in the computational basis [3]), but it is also a part of several quantum algorithms [1].

Shor's algorithm begins by preparing two quantum registers. The first register holds the number to be factored (denoted as "a"), while the second register is allocated for the "period-finding" subroutine. The QFT serves as a quantum counterpart to the classical discrete Fourier transform and holds a crucial role in converting the periodic behavior of a function into the frequency domain.

2.2.4 Period-Finding Subroutine

This phase exploits the quantum parallelism and superposition capabilities of quantum computers. By performing a series of modular exponentiations, the algorithm creates a superposition of states [3], with each state corresponding to a distinct value of the function's period [3]. This is achieved through the efficient implementation of a modular exponentiation gate, which computes $a^x \mod N$ for varying x values. The resulting superposition encodes the period of the function, a key factor in the factorization process.

2.2.5 Quantum Measurement

Quantum measurement collapses the superposition of period states into a single value. The measured value provides an approximation of the period of the function. The probabilistic nature of quantum measurement means multiple iterations of the algorithm may be required to achieve a prominent level of certainty in the measured period.

2.3 Steps to Shor's algorithm

1. *Choose a Number to Factor:* Start by selecting the number you want to factor. Let's call this number N.
2. *Find a Suitable 'a'*: Next, we need to find a random positive integer $'a'$ that is less than $'N'$ and relatively prime to $'N'$. In other words, $'a'$ should not share any common factors with $'N'$ other than 1.

$$2 < a < N - 1, \ \gcd(a, N) = 1 \tag{6}$$

3. *Quantum Period Finding:* This is the core principle of Shor's algorithm. We use a quantum computer to find the "period" of a specific function. The period is the smallest positive integer 'r' such that:

$$a^r \mod N = 1 \tag{7}$$

   This step involves using the Quantum Fourier Transform.
4. *Classical Post-Processing:* After obtaining the period 'r' using the quantum computer, we need to perform some classical computations to extract useful information from it. The period 'r' has a mathematical relationship with the prime factors of 'N'.
5. *Extract Factors:* Now, we use the periods 'r' and 'a' to calculate potential factors of 'N' using some mathematical formulas. If everything goes as planned, these calculations will give us the prime factors of 'N'.

3 Challenges in Quantum Cryptanalysis

3.1 Error Correction and Fault Tolerance

Decoherence is a process in which the Qubit loses the quantum information in time, due to interactions with the environment [3]. The problem of decoherence only increases as quantum computing sizes increase [1]. The more Qubits and quantum gates a system has, the more susceptible it is to decoherence [1]. That is since there are more components that can interfere with each other [1]. At some point, decoherence in a quantum system is inevitable [1]. The issue of controlling decoherence is closely related to the issue of quantum error correction, both decoherence and error correction affect the reliability of quantum computations [1]. Quantum computers are inherently susceptible to errors due to their reliance on fragile quantum states. Error correction and fault tolerance mechanisms have emerged as pivotally important subjects to address this vulnerability and enhance the reliability of quantum computations.

3.2 Addressing Errors in Quantum Computations

The more the physical Qubits are, the more error-prone the circuit is [3]. If the probability that an individual Qubit (or a quantum gate/preparatory circuit/measurement circuit) may fail is p, then the probability of an encoded operation failing is $c * p^2$, where c is a constant [3]. Quantum error correction (QEC) techniques have emerged as essential strategies for maintaining the coherence of Qubits and enabling fault-tolerant quantum computation [3]. QEC involves encoding logical Qubits into larger quantum codes, allowing for the detection and correction of errors without disturbing the encoded information [1]. By employing error correction, quantum computations can be executed more reliably, paving the way for the successful implementation of algorithms like Shor's algorithm for Quantum factorization.

3.3 Cryptanalysis of RSA

The most widely used of the public key cryptosystems is the RSA cryptosystem [1], underpinning secure communication and digital signatures. Cryptanalysis of RSA involves several approaches aimed at uncovering weaknesses in its mathematical foundation. The few approaches towards breaking RSA are:

1. *Brute Force Factorization*, The most straightforward approach involves attempting to factor the large modulus "N" into its two prime factors, p and q. As the size of the primes increases, the difficulty of this task grows exponentially, making it infeasible for sufficiently large keys.
2. *Mathematical Attacks* Techniques like Pollard's Rho, Quadratic Sieve, and the General Number Field Sieve focus on finding vulnerabilities in the mathematical structure of RSA. They aim to efficiently factorize the modulus to derive the private key.
3. *Chosen Ciphertext Attack (CCA):* This attack involves an adversary gaining access to a decryption oracle, where they can submit chosen ciphertexts for decryption. By observing the corresponding plaintexts, attackers aim to gain information about the private key. Properly implemented RSA schemes using padding techniques like *OAEP (Optimal Asymmetric Encryption Padding) or PKCS#1 v1.5* help mitigate these attacks.
4. *Timing Attacks:* Timing attacks exploit variations in the time taken for cryptographic operations. RSA operations may take different times depending on the input data. Analyzing these timings might reveal information about the private key. Countermeasures involve implementing constant-time algorithms to ensure consistent execution times.

4 State-of-the-art Shor's Algorithm Implementation

4.1 Quantum Hardware Progress

The advancement of quantum computing technologies has led to the emergence of various state-of-the-art quantum computing platforms, each with its own unique approach to harnessing the power of quantum mechanics for computation.

4.1.1 Superconducting Qubit-Based Systems

Superconducting Qubits are leading candidates in the race to build a quantum computer capable of realizing computations beyond the reach of modern supercomputers [8]. Superconducting circuits are manufactured using a multi-step additive and subtractive fabrication process involving *lithographic patterning, metal deposition, etching, and controlled oxidation of thin, two-dimensional films* of a superconductor such as aluminum or niobium [8]. Superconducting Qubits are made such that they lose all electrical resistance when they are cooled to extremely low temperatures. The current superconducting circuits have a long coherence time in the order of milliseconds with gate periods (time intervals during which quantum logic gates execute operations on Qubits) close to tens of nanoseconds [3], providing us the opportunity to experiment quantum algorithms on real quantum devices [3]. However, challenges related to Control and high coherence in medium-scale devices [8], Improving Qubit connectivity [8], and Improved gate fidelity [8], warranting further research to address these limitations.

4.1.2 Trapped Ion Quantum Processors

Trapped ion quantum processors utilize the unique properties of ions confined in electromagnetic traps to implement Qubits and perform quantum operations [3]. The Qubit value is stored as an electronic state for each ion [1]. This of course requires stable ions [1]. The coherence time of the trapped-ion Qubits is 50 s and has a 99.97% error rate in 150,000 operations [3]. These are good numbers for trying out various quantum algorithms like Shor's algorithm and Grover's Algorithm.

Trapped-ion Qubits are constructed as Micro-circuits using the same processes used to make silicon-based integrated circuits, employing diode lasers and electrodes. The Challenge with this modality is that it employs too many lasers, and Qubit coupling is difficult. The sources of noise are the heating of the Qubits, environmental noises, and instability of the laser fields.

4.2 Performance Evaluation of Shor's Algorithm on Existing Quantum Devices

The successful implementation of Shor's algorithm for RSA breaking relies heavily on the capabilities and characteristics of the underlying quantum computing hardware. To manifest the potential of this algorithm in practical scenarios, a rigorous performance evaluation was carried out on a diverse set of representative quantum devices from various platforms.

The success of Shor's algorithm relies on the accurate and simultaneous measurement of quantum states. Any errors or deviations during this measurement phase can affect the algorithm's outcome. Fig 1. Shows the performance evaluation of classical computational methods compared to quantum computational methods.

5 Challenges and Future Directions

As Quantum Computers will be making a significant impact in the next few decades for humanity, harnessing its disruptive potential will take time until scalable systems are available. The future development of Quantum Processors, Quantum Error-Correction techniques, and Superconducting and Ion-trapped Qubit systems would be some of the most relevant questions about the potential impact on existing cryptographic systems. It is likely that one of the major applications of quantum computers in the future will be performing simulations of Quantum Mechanical systems too difficult to simulate on a classical computer, a problem with profound scientific and technological implications [9]. Much more work is needed to build post-quantum systems that are widely deployable while at the same time inspiring confidence [10].

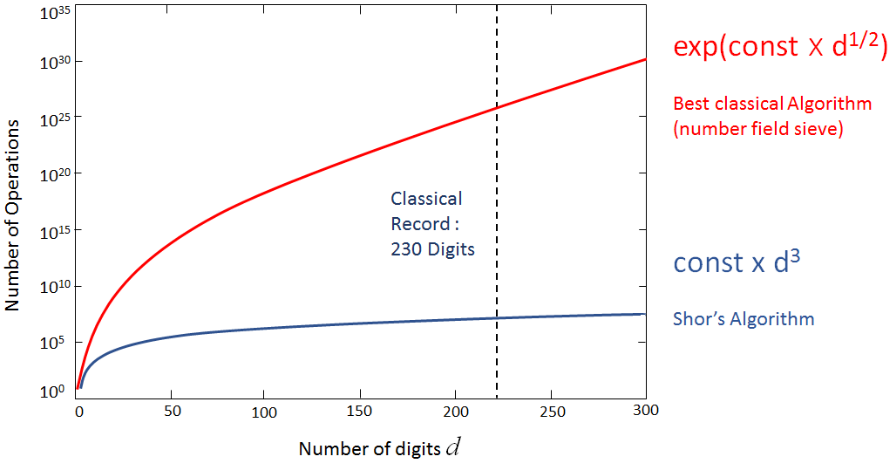

Figure 1: *(Logarithmic Scale) The evaluation was completed by comparing the results obtained from the quantum devices with those from classical factorization methods, particularly the General Number Field Sieve (GNFS), comparing the number of digits factorized with their respective number of operations performed. This analysis aimed to highlight the quantum speedup achieved by Shor's algorithm and to identify scenarios in which the quantum approach outperformed classical methods. [IBM Quantum]*

As quantum computers become more stable and widely used, there will eventually need to be an alteration of the TLS (Transport Layer Security) standard to accommodate quantum-safe cryptography [1]. The awareness and adoption of post-quantum cryptographic solutions are pivotal in ensuring the robustness of cryptographic systems against quantum-based attacks. The problem with cybersecurity is that it has already been proven that these mathematical problems can be solved with quantum computers [1]. Lattice-based Cryptography and Multivariate Cryptography are diverse types of cryptographic algorithms that have been posited as quantum resistant [1].

As the future of quantum computing is expanding its horizon, the classical systems that exist today might not be fit for a world that is quantum-powered, therefore, the current need for quantum-safe cryptography is of utmost importance. As we move towards more efficient and error-free quantum systems and highly developed quantum error correction techniques the need for quantum-safe cryptosystems rises. Lattice-based cryptography and multivariate cryptography might be the beginning of another era of fast and secure communications for a quantum-powered world.

6 Conclusion

According to the mathematical analysis of the efficiency of Shor's algorithm performed previously in this paper, this ratio of the time complexities helps us to quantify the speedup achieved by Shor's algorithm over GNFS. As N (the number required to be factorized) grows larger, the exponential terms in GNFS become dominant, leading to a considerable difference in the execution times between the two methods. The exponential speedup of Shor's algorithm increases its potential to drastically reduce the time required for factoring large semi-prime numbers. The mathematical analysis of the execution times of Shor's algorithm and GNFS provides a quantitative understanding of the exponential speedup offered by Shor's algorithm in comparison to classical factorization methods. Thus, it makes it clear that a practically powerful quantum computer can break any length of RSA encryption in a feasible amount of time, even the best-known classical theories to factorize numbers are exponentially less powerful when we compare them with the time complexities of algorithms like Shor's Algorithm. The current state of Quantum computer Hardwar is not up to the standard to beach RSA encryption of considerable difficulty but the time might be close enough when our hopes for Quantum-safe Cryptosystems may start from Lattice-based and multivariate cryptography, which might be the beginning of another era of fast and secure communications for a quantum-powered world.

Authors

Siyon Singh *is a Junior at Foundation Academy, IIT Patna, whose academic research pursuits gravitate towards the profound complexities inherent in Quantum Physics and Computation.*

Email: siyonsingh2007@gmail.com

Eric Sakk *is an Associate Professor in Computer Science at Morgan State University in the domain of Dynamical systems, machine learning, system theory, and bioinformatics.*

Email: Eric.Sakk@morgan.edu

References

[1] W. C. Easttom II, *Quantum computing fundamentals.* Addison-Wesley Professional, 2021.

[2] M. J. Nene and G. Upadhyay, "Shor's algorithm for quantum factoring," in *Advanced Computing and Communication Technologies: Proceedings of the 9th ICACCT, 2015.* Springer, 2016, pp. 325–331.

[3] V. Kasirajan, *Fundamentals of quantum computing.* Springer, 2021.

[4] A. Albuainain, J. Alansari, S. Alrashidi, W. Alqahtani, J. Alshaya, and N. Nagy, "Experimental implementation of shor's quantum algorithm to break rsa," in *2022 14th International Conference on Computational Intelligence and Communication Networks (CICN).* IEEE, 2022, pp. 748–752.

[5] V. Bhatia and K. Ramkumar, "An efficient quantum computing technique for cracking rsa using shor's algorithm," in *2020 IEEE 5th international conference on computing communication and automation (ICCCA).* IEEE, 2020, pp. 89–94.

[6] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM review*, vol. 41, no. 2, pp. 303–332, 1999.

[7] T. Monz, D. Nigg, E. A. Martinez, M. F. Brandl, P. Schindler, R. Rines, S. X. Wang, I. L. Chuang, and R. Blatt, "Realization of a scalable shor algorithm," *Science*, vol. 351, no. 6277, pp. 1068–1070, 2016.

[8] M. Kjaergaard, M. E. Schwartz, J. Braumüller, P. Krantz, J. I.-J. Wang, S. Gustavsson, and W. D. Oliver, “Superconducting qubits: Current state of play,” *Annual Review of Condensed Matter Physics*, vol. 11, pp. 369–395, 2020.

[9] M. A. Nielsen and I. L. Chuang, *Quantum computation and quantum information.* Cambridge university press, 2010.

[10] D. J. Bernstein and T. Lange, “Post-quantum cryptography,” *Nature*, vol. 549, no. 7671, pp. 188–194, 2017.