M.Sc. Mathematics

Sukhpal

2022

Quantum Communications and Cryptography

A DISSERATION

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE AWARD OF THE DEGREE

OF

MASTER OF SCIENCE

IN

MATHEMATICS

Submitted by:

Sukhpal

2K20/MSCMAT/30

Under the supervision of

Mr. Rohit Kumar

DEPARTMENT OF APPLIED MATHEMATICS

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi-110042

MAY, 2022

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi-110042

DECLARATION

I, Sukhpal, 2K20/MSCMAT/30 student of M.Sc. Mathematics, hereby declare that the project Dissertation titled "Quantum Communications and Cryptography" which is submitted by me to the Department of Applied Mathematics Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of the degree of Master of Science, is original and not copied from any source without proper citation. This work has not previously formed the basis for the award of any Degree, Diploma Associateship, Fellowship or other similar title or recognition.

Place: Delhi **SUKHPAL**
Date:

DEPARTMENT OF APPLIED MATHEMATICS

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi-110042

CERTIFICATE

I hereby certify that the Project Dissertation titled " Quantum Communications and Cryptography " which is submitted by Sukhpal, Roll No. 2K20/MSCMAT/30 [Department of Applied Mathematics], Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of the degree of Master of Science, is a record of the project work carried out by the students under my supervision. To the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

Rohit Kumar

Digitally signed by Rohit Kumar
DN: C=IN, OU=Department of Applied Mathematics, O="Delhi Technological University, Delhi", CN=Rohit Kumar, E=rohitkumar@dtu.ac.in
Reason: I am the author of this document
Location: your signing location here
Date: 2022.05.05 19:38:45+05'30'
Foxit PDF Reader Version: 11.2.1

Place: Delhi
Date:

MR. ROHIT KUMAR
SUPERVISOR
ASSISTANT PROFESSOR
DEPARMENT OF APPLIED MATHEMATICS
DELHI TECHNOLOGICAL UNIVERSITY
BAWANA ROAD, DELHI-110042

ABSTRACT

Computer science is fundamentally linked to the physical world.

The machines we use to access the Internet, the Java programs written by a student in an introductory computer science class, and indeed everything we generally refer to as a "computer"is based on a set of physical assumptions. Unfortunately, the underlying physical assumptionson which nearly all modern computers rely are outdated. The twentieth century saw the rise of quantum mechanics, revealing strange phenomena including superposition and entanglement, whilewe are still stuck with computers based on classical physics. The new and incredibly rich field of quantum computation and information seeks to harness the power of quantum mechanics in order to develop significantly more powerful computational techniques .

The classical picture of the world is intuitive. We think of electric current as the motion of electrons across a voltage, and of electrons as discrete particles. They cannot be in more than one state simultaneously, and they have a definite location, momentum, energy and time. Quantum mechanics is more bizarre. Quantum computers open up a slew of new cryptographic possibilities, including key distribution, confidentiality, integrity, and non-repudiation. Quantum cryptographic techniques differ significantly from conventional cryptography due to the destructive and probabilistic character of quantum measurements, as well as the no-cloning theorem, which forbids the duplication of quantum states. Although a vast number of these methods have been presented, there has been little comparative research. In order to provide an overview of the present state of the subject, we conduct a study of existing protocols with a focus on practical applications.. We also show how to make a quantum cryptography algorithm that provides confidentiality, integrity, and non-repudiation. Additionally, we generalise our implementation method to give a template for converting algorithms from a research article to a quantum programming environment.

ACKNOWLEDGEMENT

The satisfaction of successful completion of any task would be incomplete without mentioning the people who made it possible and whose constant guidance and encouragement crown all of my efforts with success. I express my deep sense of gratitude to my parents, my brother and my guide Mr. Rohit Kumar, Assistant Professor, Department of Applied Mathematics, Delhi Technological University for her inspiration, constant assistance, valuable suggestions, sympathetic advice, fruitful conversation and unparalleled encouragement made throughout the course of study, without which this piece of work would not have taken its present shape.

SUKHPAL

Table of Contents

List of Tables

List of Abbreviations

(3DES): Triple Data Encryption Algorithm 8
(AES): Advanced Encryption Standard 17
(AQS): Arbitrated Quantum Signatures 5
(BQP): Bounded-Error Quantum Probabilistic Polynomial 11
(COW): Coherent One Way 15
(DNA): Deoxyribonucleic Acid 5
(DPS): Differential Phase-Shift 15
(DSQKD): Decoy-State QKD scheme 4
(ECB): Electronic Code Book 10
(ECC): Error Correcting Code 22
(ECC): Excise Control Code 30
(IBM): International Business Machines 20
(MDI-QKD): Measurement Device Independent Quantum Key Distribution 15
(MitM): Man in the Middle Attack 9
(NIST): The National Institute of Standards and Technology 11
(NSA): National Secuirty Agency 9
(NTRUEncrypt): N-th degree Truncated Polynomial Ring Units 18
(PNS): Photon-Number Splitting 8
(QAA): Quantum Adiabatic Algorithm 12
(QBER): Quantum Bit Error Rate 11
(QKD): Quantum Key Distribution 7
(QKP): Quantum Key Pool 4
(QMACs): Quantum Message Authentication Codes 5
(SHA): Secure Hash Algorithm 8
(TF-QKD): Twin-Field QKD 15

CHAPTER 1

INTRODUCTION

The world is on the verge of a new computing revolution because to quantum computers. Quantum computers will be able to tackle complicated problems in minutes that a classical computer would take thousands of years to solve using the unique features of subatomic particles. Many of these issues, such as the prime factorization problem at the heart of internet security, are basic to current cryptography.While quantum computers currently lack the intelligence required to crack today's encryption methods, it is only a matter of time before they do. This research intends to investigate the consequences of quantum computing on cryptography in order to stay ahead of this looming threat and protect sensitive communications ranging from internet browsing to military commands.

Messages must be guaranteed delivery, not tampered with, and authenticated and correct in very sensitive communications systems, such as those used to begin a military operation. Non-repudiation is a significant concern since these systems must not allow for misleading communications, which might result in widespread death in the worst-case scenario. In a world when quantum computers are common and capable, the purpose of this study is to determine the optimal ways for a crucial system to convey confidential information with a guarantee of integrity and non-repudiation. Quantum key distribution, quantum digital signatures, and quantum-resistant encryption algorithms executed on a conventional computer are examples of cryptographic processes.

1.1 Background and Problem Context

Data may be sent practically quickly from one side of the world to the other in the era of 5G high-speed internet surfing. The disadvantage of this technical convenience is that the confidential information shared by both parties in a communication might be easily exposed to a malevolent third party. Fortunately, current cryptographic methods are sufficient to secure users' privacy because the most majority of them are impossible to crack using regular computers' processing capability. When quantum computers become available, however, practically every currently employed encryption technique will be susceptible, resulting in

enormous data leaks. As a result, it’s critical to comprehend quantum computers and why they’re capable of destroying conventional cryptography.

A quantum computer is a physical device that can overcome many of the constraints of traditional computers. Quantum computers use quantum mechanics principles to perform high-speed mathematical and logical processes in order to store and process qubit data. Quantum computers have their own basic unit, the quantum bit, which distinguishes between zeros and ones, or bits, by turning on and off logical gates on an integrated circuit. The classical bit states of 0 and 1 are substituted by two quantum states in these quantum bits, or qubits, $|0\rangle$ and $|1\rangle$. The two orthogonal polarisation directions of a photon, the spin directions of an electron in a magnetic field, the two directions of a nuclear spin, or the multiple energy levels of an electron in an atom can all be represented in a quantum computer.

The fact that qubits can exist in a superposition of two logical states is a more significant distinction between bits and qubits. "Unlike classical bits, quantum bits can be in a superposition state that encodes both 0 and 1." There is no adequate classical explanation for superpositions: a quantum bit expressing 0 and 1 cannot be interpreted as being ’between’ 0 and 1 or as a hidden unknown state representing either 0 or 1 with a fixed probability." Qubits can also become entangled, causing measurements of one to have an impact on others. As a result of these qubit characteristics, quantum computers may process information differently and efficiently tackle a new class of problems than traditional computers.

1.2 Motivation

Encryption is used in all aspects of modern life, from internet browsing to medical gadgets, wireless vehicle keys to nuclear control systems. Quantum computing poses a danger to practically all present encryption algorithms in some form, either by significantly lowering key strength or by completely destroying the algorithm. As a result, grasping the significance of quantum computing and novel cryptographic protocols is crucial, with implications ranging from personal privacy to national security.

Unlike in the previous century, fighting between countries is no longer just a matter of employing modern weapons or gaining a geographical advantage to suppress or destroy the adversary. Instead, modern combat is dominated by information transmission and

manipulation, sometimes known as information warfare. It is critical in this instance to send information between the agent and the commander in a timely and secure manner. For example, messages in very sensitive communications systems, such as those used to launch a military attack, must be guaranteed delivery, untampered with, and authenticated and correct. False messages must not be allowed, as they might result in enormous leaks of state secrets or, in the worst-case scenario, widespread loss of life, hence non-repudiation is a major problem.

1.3 Goals and Research Questions

As previously stated, the purpose of this project is to find the best means for a crucial system to convey confidential information while maintaining secrecy, integrity, and non-repudiation in a world where quantum computers are common and capable. Quantum key distribution, quantum digital signatures, and quantum-resistant encryption algorithms run on a conventional computer are examples of cryptographic processes. As a result, we've asked the following six questions, which we'll try to address in this paper:

- What impact will quantum computing have on today's most widely used cryptographic methods for hashing, symmetric, and asymmetric encryption? Which of these protocols, or groups of protocols, should quantum or post-quantum solutions be used to replace?

- What quantum protocols are the most effective for generating and distributing cryptographic secret keys?

- What are the most promising ways for maintaining data confidentiality at rest and in motion, either quantum or a hybrid of classical and quantum procedures?

- What are some appropriate transmission protocols for confirming the integrity of quantum data?
- What quantum cryptography algorithms are the most effective at assuring message authenticity and non-repudiation?

- Is there a systematic procedure for translating a theoretical quantum algorithm into a testable implementation?

CHAPTER 2

RELATED WORK

2.1 Contemporary Quantum Communication and Cryptography

After studying a number of academic papers and official publications, we discovered that contemporary quantum technology research covers a wide range and yields substantial results. However, not all of those resources are important; some of them may work against our purpose and are difficult to achieve using current technology. As a consequence, we divided various potentially useful resources into five groups based on the project's purpose.

2.1.1 Quantum Key Distribution

Shor's algorithm poses a danger to contemporary encryption, despite the fact that QKD protocols have been shown to ensure unconditional communication security. Some researchers compared results while a third party eavesdropped on QKD protocols including BB84, B92, and BBM92 to see how many keys could be received and how many errors could occur during transmission. Finally, they discover that if we can properly implement QKD on a quantum computer, the unconditional quantum communications security may be demonstrated.

The current rate of producing keys for QKDN is slow. As a result, further research has looked into a concept known as Quantum Key Pool (QKP), which aims to offset the inefficiencies of key production by storing generated keys. However, because the QKP needs to keep keys for a time, the security of QKD will be compromised, and the fundamental performance of QKDN will be affected.

Although there is still a gap between the current QKD system and the ideal one, some scientists have overcome this obstacle by implementing the Decoy-State QKD scheme (DSQKD) to increase the security and performance of QKD transmission. The pre-request is the data exchanges between two nodes in the DSQKD protocol, however, are limitless. It's difficult to achieve with the real-world QKD system's limited data interchange rate.

The secure key rate scale can be nearly doubled in perfect TF-QKD, allowing it to be used for

significantly longer-distance transmission than regular QKD. In this situation, TF-QKD not only protects data confidentiality, but it may also be employed over a far greater distance. However, building mode matching systems to complete the first interference of two types of lasers is quite challenging.

2.1.2 Quantum Non-Repudiation

Because digital signatures are important for verifying a message's integrity and authenticity, some researchers devised a scheme that combines a dynamic map based on quantum dots, a permutation and substitution scheme like AES, and DNA coding to create a quantum digital signature with a high level of security as long as the signature is long enough. To put it another way, two parties might use a quantum computer to implement these digital signatures by providing a dynamic quantum system's control parameter and critical points, as well as some beginning point in phase space.

Both classical and quantum messages can benefit from quantum signatures. It's useful to be able to sign a communication with just a single qubit and a trustworthy third party. The key forging is impossible with Arbitrated Quantum Signatures (AQS), but full non-repudiation is not.

Quantum cryptography approaches may be able to achieve greater integrity, data origin authentication, and non-repudiation. Quantum Message Authentication Codes (QMACs), for example, are thought to be superior to traditional message authentication methods by some. However, researchers discovered that information-theoretically secure message authentication performed better in classical cryptography, and that several known QMAC techniques are inferior to their classical counterparts after conducting several tests.

Few researchers have discovered that binary classical messages may be validated using a set of QMAC protocols that, by employing a single qubit as the authentication key, can successfully authenticate messages with a chance of forgery of less than one. This QMAC system also allows for key reuse, albeit security is not guaranteed.

2.1.3 Integrity and Post-Quantum Security

It can strengthen attacks against hash functions, key recovery in multi-user situations, and

collision attacks on block cypher operating modes by using an amplitude amplification technique, quantum collision, and multi-target preimage search algorithm. This approach can be used as a foundation for more advanced cryptanalysis. Furthermore, the proposed approach reduces the time complexity of existing algorithms while needing less quantum memory. Comparisons of new and current algorithms are done under a variety of scenarios including quantum memory availability, with the conclusion that this new algorithm is preferable unless quantum memory becomes as inexpensive as classical memory and parallelization becomes impossible to achieve.

2.2 Quantum computing's effects on classical cryptography

Quantum computing promises a rapid increase in computational capacity. The computational power of quantum computing is proportional to the system's size. Computational parallelism is the term for this type of growth, and it is this increase that makes quantum computing a possible next stage in computer evolution. Quantum computing by itself does not pose a threat to encryption or communication, but when it is paired with quantum algorithms, it poses a risk. Quantum key distribution was intended to protect asymmetric key distribution from the threat that quantum computing brings to asymmetric encryption. Quantum computing has the potential to jeopardise present hashing standards and digital signatures, hence quantum non-repudiation was created to assist preserve them. As quantum computing advances, it poses a bigger threat to traditional encryption, necessitating the employment of quantum algorithms to mitigate these threats.

Computers today learn how to do something by attempting every conceivable combination and selecting the best one. Quantum computers can try every combination at the same time due to superposition; a quantum computer can be both the correct and incorrect paths at the same time. Quantum computing makes use of ambiguity in its state, and it can be both right and wrong. You can measure the response without collapsing the quantum state by using entanglement. Two particles are entangled when they are related but physically separate. You can gauge the state without collapsing the wave function by using one set of particles.

2.2.1 Shor's Algorithm Impact on Asymmetric Cryptography

Shor's Algorithm, invented by Peter Shor, is a quantum algorithm. Shor's Algorithm is a quantum factoring algorithm that works in polynomial time. Shor's Algorithm is too resource

expensive to execute on a regular computer; it must be run on a quantum computer to receive the full effect of Shor's Algorithm. Quantum bits, also known as qubits, are a unit of quantum information that Shor's Algorithm requires. Using Shor's Algorithm, a polynomial-time factorization problem that can compromise the security of RSA, elliptic curve Diffie-Hellman, and most other contemporary asymmetrical encryption systems with enough qubits. Shor's Algorithm requires more qubits than can currently be manufactured in order to pose a threat to current asymmetrical encryption. This isn't to say that asymmetric encryption isn't secure.

Shor's Algorithm researchers developed lattice and ring-based encryption to protect conventional computer against the future possibility of quantum computing. Lattice and ring-based encryption are based on mathematical algorithms that have been studied since the 1980s and have yet to be broken. Quantum computing can encrypt data in transit using quantum key distribution. Bennett and Brassard suggested quantum key distribution using BB84 in 1984.

BB84 can use multiplexing and go up to 200 kilometres; without multiplexing, it can travel up to 240 kilometres. The tools are still in the early stages of development. Protocol E91 supports multiplexing and can carry data over distances of up to 200 km and 240 km without it. E91 protocol Developed by Arthur Ekert in 1991, Protocol E91 prevents attackers from guessing results. Protocol E91 consumes an excessive amount of resources. MDI-QKD has the best range of 404 km, however it requires unique transmission channel setup.

Asymmetric encryption is now utilised to safeguard existing communication systems and is a secure method of sending data over the Internet. With Quantum Computing, this could all change, hence Quantum Key Distribution (QKD) is being developed to help secure future communication. Quantum Key Distribution is a type of encryption that uses a one-time pad. "As we all know, One-Time-Pad is the safest way to communicate between two network nodes, so Quantum Key Distribution (QKD) is using it to create a much safer network environment called QKDN."

2.2.2 Grover's Algorithm Impact on Symmetric Cryptography

Grover's quantum search algorithm is a sophisticated quantum computing algorithm that may be used to solve the challenge of finding a specified object among N unclassified things. The classical algorithm, in particular, can only search one after the other until it discovers the thing

it seeks. On average, this algorithm has $O(N)$ complexity, but Grover's quantum algorithm has only $O(\sqrt{N})$ complexity.

Grover's technique, for example, can minimise the time necessary for brute force attacks. A quantum computer can degrade the security of keys by a factor of $O(\sqrt{n})$ for public key encryption algorithms like AES and 3DES, making brute forcing a 256-bit key equal to brute forcing a 128-bit key with a regular computer. Grover's approach minimises the time required for a collision attack while also weakening the hash function's security. The security strength of SHA256 was reduced from 128 bits to 80 bits or less with the quantum computer, and the security strength of SHA384 was reduced from 192 bits to 128 bits.

Despite IBM's dissenting opinion that it would take only two days instead of 1,000 years, Google used a 53-qubit quantum computer to demonstrate that quantum computing systems have some unique capabilities that can outperform traditional computers (solving a problem that would take supercomputers 1,000 years to solve in 2.30 minutes). However, it has essentially demonstrated that quantum computers outperform ordinary supercomputers on specific challenges, allowing humanity to travel to previously unexplored boundaries.

2.3 Attacks against quantum cryptography techniques

2.3.1 Photon Splitting Attack

Alice employs a single photon source, which is a crucial assumption in the ideal BB84 protocol. However, preparing a single photon source in the actual system is problematic, thus a weakly coherent light source, which may be generated by attenuating the laser light source, is commonly employed. The photon number distribution of a weakly coherent light source follows the Poisson distribution, with a significant multi-photon component. Eve may eavesdrop on multi-photon components via photon-number splitting (PNS) attacks.

The following is the core principle of PNS attack: Eve intercepts Alice's weak coherent pulse and uses quantum non-destructive measurement to obtain the photon number information. All interceptions are no longer sent to Bob for the single-photon phase; for the multi-photon part, Eve takes one photon and stores it in its own quantum memory before sending the remaining photons to Bob through a low loss or even no loss channel (ideally). After Bob uploads his

measurement basis vector, Eve uses the same basis vector to measure the photons in his quantum memory. Then, using Alice's basis vector information, Eve performs the identical data post-processing operation as Bob, resulting in Eve having the exact same key as Bob.

2.3.2 Denial of Service

"Quantum key distribution increases the risk of denial of service," the NSA declared in a 2020 assessment. The theoretical basis for QKD security claims, sensitivity to an eavesdropper, also shows that denial of service is a substantial danger for QKD."

The quantum network, like the classical communication network, is vulnerable to denial of service attacks. In a nutshell, a DoS assault, also known as a denial of service attack, refers to hackers attempting to breach the target system or server in order to prevent it from working, which is one of the most prevalent hacking methods. To carry out a DoS attack, hackers typically send a large number of malicious requests via the network in order to overload the target resources until they crash, preventing other lawful users from accessing them. DoS attacks are known to cause significant financial losses in a variety of sectors, including governments and businesses.

2.3.3 Man in the Middle Attack

QKD creates the keys for the encryption technique to ensure the communication's privacy, but it cannot provide an authentication mechanism to the transmission's originator. From a technological standpoint, QKD's security is degraded since it cannot prevent a man in the middle (MitM) attack. The (NSA) study presents major concerns that cannot be overlooked or ignored.

This scenario could be found in the quantum transmission layer. Each qubit is sent with a polarisation of 0° and 45°, indicated as $+$ and $\times$, respectively, by the sender, while each qubit is received with a polarisation of $+$ and $\times$ by the receiver. Over an insecure channel, like as the Internet, the receiver informs the sender of its sending bases, and the sender indicates which parts are correct. In this situation, the sender and receiver will ignore any qubits with wrongly set listeners. The sender and receiver then compare half of the remaining qubits, and if there is an error, it means that there is another listener present in addition to the receiver, interfering

with the photon's vibrational orientation. The bits are discarded if there are no errors, and the remaining bits are utilised as the key. If an eavesdropper is present, the last check operation will fail because he will change the state of the initial photon, causing the qubits to be incorrect half of the time.

The MitM attacks scenario could now be considered. The initiator of a MitM attack is a more powerful actor than the listener, as he not only has access to all Internet communication packets between the two parties, but he also has the ability to modify those packets as he wishes. As a result, he can present himself to the sender as the recipient and to the receiver as the sender. Once the attack begins, the middleman chooses one of the + and × bases at random at the start of the communication, tampers with the message so that the sender receives the receiver's bases that are identical to the middleman's bases, and tampers with the message so that the receiver receives the correct pattern, which is the qubits for which the middleman and receiver have the same bases. The sender and the middleman both retain the same bits in the final verification procedure, and the middleman also knows the bits that the receiver holds.

2.4 Cryptography Survey Methodology

Two additional groups have carried out comparable studies on a variety of cryptographic protocols. Jorstad and Smith seek to answer the question "Can a standard objective framework for the measurement and specification of cryptographic algorithm strength be created" in a study published in 1997. Their approach was based on known properties of previous algorithms and how they might be compared. They concentrated almost solely on civilian encryption methods, both symmetric and asymmetric, intended for commercial usage and working in Electronic Code Book (ECB) mode. However, because they only used algorithms that existed in 1997, their work is relatively old, but their recommended categorization method is relevant. Type (symmetric or asymmetric), functions (secrecy, integrity, etc.), key size, rounds, complexity, attack, and strength are all covered by their suggested classification scheme.

Khan et al. published a similar study in 2020 that focused solely on QKD techniques. Most modern QKD protocols have not been thoroughly compared in terms of security; however, it is vital to verify the deviation between theoretical elements and real-world usage, particularly in terms of simulation and implementation. The authors of the research present a straightforward

quantitative comparison of 11 distinct QKD methods across six different parameters, as well as a simulation of the BB84 and 2-dimensional KBM09 protocols. Even though their experimental comparison is limited to evaluating the Quantum Bit Error Rate (QBER) reliability of two protocols, their findings provide useful guidance on how protocols can be compared both theoretically and experimentally, and serve as a foundation for the classification of quantum cryptography schemes.

2.5 Post-quantum cryptography

The National Institute of Standards and Technology (NIST) is leading a project called Post Quantum Standardization (PQS) that attempts to establish new techniques for dealing with quantum computer vulnerabilities. The PQS project is nearing completion and should be completed in two years.

SSH, VPN, IPSec, SSL/TLS, and other security protocols must all be improved in order to make the transition to quantum secure computing a reality. These protocols must be coupled with current protocols, but they must also bring an additional layer of security to protect against quantum attacks. This modification will have an influence on asymmetric encryption and key generation techniques, while symmetric cryptography algorithms will need to raise their key size. There is also a performance and bandwidth impact. In order to align and migrate with these new algorithms, hardware providers will need to change their hardware.

We must also guarantee that the new algorithms do not fall into the bounded-error quantum probabilistic polynomial (BQP) complexity category. BQP stands for "bounded error quantum polynomial time," which was defined by computer scientists Ethan Bernstein and Umesh Vazirani in 1993. They describe this category as all of the yes-or-no decision issues that a quantum computer can solve. To put it another way, there is a quantum method for a BQP problem that runs in polynomial time and has a high probability of getting the correct answer. The probability of getting the incorrect response should be smaller than $1/3$ in any given case. BPP, which stands for bounded-error, probabilistic, and polynomial time, can alternatively be thought of as the quantum computer counterpart of BQP.

2.6 Types of Quantum Computing

The Hamiltonians, a function that describes all energy in a system, are optimised using

adiabatic quantum algorithms. The Hamiltonian utilises an operator to represent a system's energy. In both kinetic and potential energy, the Hamiltonian corresponded to total energy. The quantum adiabatic algorithm (QAA) was created to address quantum computer optimization concerns. The combinatorial optimization issue will be optimised by QAA in a dedicated device. When in the ground state, the combinatorial optimization problem is solved by evolving adiabatically, where only energy is transmitted. In the circuit model, adiabatic quantum computing can be as powerful as non-stoquastic Hamiltonians. This indicates that the eigenvalue gap is as simple as possible for a many-body Hamiltonian. QAAs provide quantum speedup and can solve some of the problems associated with qubit quantum computing, such as the need for fewer qubits to crack asymmetric encryption. The problem with adiabatic quantum computing is that it is intrinsically unstable as the number of qubits increases, which limits its future applications. QAAs are excellent for tackling satisfiability and combinatorial search problems.

The circuit model of quantum computing is now the most popular type of quantum computing. To handle its qubits, circuit model quantum computing uses Hilbert space, a vector space that permits defining lengths and angles, and a set of unitary quantum logic gates. To generate more sophisticated quantum operations, multiple quantum logic gates can be coupled together. Classical logic gates and quantum logic gates differ significantly in that quantum logic gates can travel backwards and forwards, whereas classical logic gates can only go forwards. Despite their one-of-a-kind character, quantum gates can be used to create a full boolean algebra and even a full quantum Turing machine.

CHAPTER 3

METHODOLOGY

In their 2020 evaluation of QKD protocols, Khan, Meraj, and Khan utilised a methodology that is very similar to ours. The primary goal of this study was to conduct a comparative examination of quantum cryptography techniques. From the literature, a number of algorithms representing various features of cryptography, such as key distribution, confidentiality, integrity, and non-repudiation, were chosen. We looked at these protocols and found key criteria that we thought may be utilised to distinguish them from one another.

Because categorising an encryption algorithm based on its verifiability would be counterproductive, algorithms were classed using distinct criteria based on their intended function. Some algorithms that perform numerous purposes, such as confidentiality and non-repudiation in the case of, are classified into multiple categories. The purpose of this investigation was to find answers to our research questions and to identify algorithm possibilities for use in a quantum simulator.

3.1 Quantum Key Distribution

Quantum Key Distribution, or QKD, is defined as the transfer of a secure key between two users utilising quantum-related technologies. The original purpose of this project was to find a proper and efficient Quantum Key Distribution protocol that would meet both performance and security requirements during key transmission between two network nodes or even more users. Because the real-world application of QKD differs from the theoretical approach, we'd like to conduct some study in order to make QKD more practical. In the table below, we've provided a high-level overview of the various QKD procedures that are either being studied by other researchers or have been tested using real quantum-related equipment.

Bennett and Brassard introduced the first QKD protocol, known as BB84, in 1984. Every bit of the secret key is encoded into the polarisation state of a single photon. Because the polarisation state of a single photon cannot be measured without destroying it, the eavesdropper will not have access to this information unless he reveals himself or resends the photon. The BB84's advantages are clear here. Eve's measurement will change the status of original qubits,

therefore it can provide a reasonable security level.

In terms of BB84's disadvantages, we recognise that when there is an eavesdropper, the transferring message will be affected, but it is still a protocol that can be implemented utilising present infrastructure. Furthermore, the basic methods for key transfer can be broken down into six steps:

1. Alice sends a random sequence of 0s and 1s qubits, with the bases × and + alternating at random.

2. Bob receives Alice's qubit sequence and alternates the measurements between bases × and + at random.

3. In a public channel, Alice broadcasts the sequence of bases used.

4. Bob informs Alice about the examples in which he was able to guess the origin bases.

5. They both compare a portion of the result to see if the error rate is above or below the required level.

6. They have both defined a random stream of bits that will serve as OTP for transmission by employing the bits of two match identical bases.

Aside from BB84, two other protocols with the same key transmission duration exist. E91 is the first, and B92 is the second. The B92 is a simplified version of the BB84 since it only employs two polarisation states (0 degrees and 45 degrees), whereas the BB84 uses four (0, 45, 90, and 135 degrees); nonetheless, the security level of the B92 is lower than that of the BB84. The E91, on the other hand, is the first protocol to use opposing measures responses between two parties, and it is based on quantum entanglement in its architecture. Despite the fact that its security level is slightly higher than BB84, it consumes far too many resources, including system, hardware, and supplements.

We also looked at two other QKD protocols that are believed to be the most secure and have a high transmission rate of long-distance qubits:

3.1.1 Measurement Device Independent QKD

The transmission distance in Measurement Device Independent QKD (MDI-QKD) is 404 km, and it is known as a straightforward solution for removing all (existing and yet to be discovered) detector side channels, especially during the installation phase. It proves that it has outstanding security and performance in theory. To be honest, when compared to the other QKD protocols, it is the easiest to construct, and its sending rate is reasonable, but it is difficult to configure using current network transmitting channels.

3.1.2 Twin-Field QKD

The longest photon transmission rate is 509 kilometres for Twin-Field QKD (TF-QKD). The detection after single photon interference is used as an effective detection event in dual-field quantum key distribution, which is measurement equipment independent. It simply requires a single detector response and does not require the standard measurement equipment's independent quantum key distribution of two photons to meet the two detectors' needed simultaneous response. In other words, during the process of the optical pulses initiated by the two users who want to establish secure connection, the information in TF-QKD is encrypted.

A single photon interference measurement from a user in the middle is then used to generate the secret key. It appears to be a viable option, but it necessitates a large number of control systems, such as a phase-lock maintenance system, a timing control system, and a polarisation system, among others.

By the way, there are still many QKD protocols we haven't looked into, such as the Differential Phase-Shift (DPS) Protocol, which can prevent PNS attacks in some ideal photon source conditions; the SARG04 Protocol, which increases the secure level against PNS attacks due to its reconciliation phase; the Coherent One Way (COW) Protocol, which is designed to counter the challenge of single photon sources; and the KMB09 Protocol, which is a more advanced protocol that can Finally, the S13 and LZWW16 protocols are enhanced versions of the BB84 protocol. As a result, we know that the BB84 is a critical QKD protocol to understand and implement because once we do, it may serve as a springboard for us to investigate the other protocols that are based on it.

3.2 Confidentiality

Research institutions such as NIST's CRSC have not explicitly specified criteria for classifying and evaluating quantum cryptosystems. The following criteria were found to be useful for classifying and evaluating selected quantum cryptosystems in terms of their capacity to maintain confidentiality:

- Target data
- Key characteristics and reusability
- Resource requirements/limitations

The data that a cryptosystem works to secure is an important consideration in evaluating its candidacy for real-world implementation. As with classical cryptosystems, the characteristics and reusability of keys used in quantum cryptosystems can serve as a primary differentiator of comparable systems and as an indicator of potential system weaknesses and vulnerabilities. Notable resource requirements or limitations for a cryptosystem's implementation are also important in considering a system's viability in real-world applications.

Few quantum algorithms have been created to provide anonymity throughout the transmission of quantum or conventional data to date. Despite the paucity of research on quantum cryptosystems, a few proposed systems stand out as prospects for proof-of-concept implementation or as a foundation for future research. Kak's Three-Stage Protocol is one of the few completely quantum solutions for data encryption over a public channel that can also be used as a simple key exchange technique. Amerimehr and Dehkordi suggested a technique that maintains secrecy during the transfer of traditional data without using a public channel while also maintaining integrity and non-repudiation. Pleşa proposes a multi-channeled, hybrid system that employs a quantum teleportation circuit for key exchange and a conventional channel for data transfer to achieve confidentiality. This hybrid system shows that quantum cryptosystems can be implemented with existing classical infrastructure in the near future. Table provides a summary of various algorithms as well as their key features.

Technical challenges regarding the ability to store qubits have limited the research and development of algorithms for encrypting quantum data at rest. Advancements in this field appear to be few and far between; even in perfect conditions, the longest period of time that

qubits can be stored before decoherence is between three and six hours.

Algorithm	**Target Data**	**Key Characteristics**	**Resource Requirements/Limitations**
Kak's Three-Stage 2006	Quantum	Single-Qubit	Quantum channel, Single-qubit data
Amerimehr and Dehkordi 2018	Classical	Length equal to message Length, Reusable	Limited message length, Algebraic ECC
Pleşa 2017	classical	single-qubit	Multi-channel; Shared, Entangled qubits

Table 1.Classification of Quantum Confidentiality Algorithms

3.3 Integrity

In traditional computing, integrity refers to ensuring that data is genuine, correct, and safe. Due to the fundamental nature of quantum computing, ensuring that the data is true and accurate can be a challenge. AES for data at rest is used to protect the integrity of classical computing against the threat of quantum computing in the future. When data is in motion lattice based encryption, NIST is still doing testing, has shown the best resistance to quantum computing. NIST is still testing different hashing algorithms. For integrity of quantum computing Zero knowledge will tell you if data has been tampered with. The nature of quantum can increase integrity, when in superposition data can't be tampered with or the wave function collapse. For data in motion QKD paired with quantum authentication encryption using a pre-shared secret key will maintain quantum computing integrity. These choices have been vetted by multiple outside research.

It's difficult to maintain honesty when the answer is neither correct nor incorrect until it's measured and studied." Quantum parallelism is the name for this process. Measuring the output states, on the other hand, will yield only one of the values in the superposition at random, while simultaneously destroying all of the other computation results." Quantum nature can also boost data integrity through the wave function; if the data is tampered with, the superposition will

alter and the wave function will collapse. Until the quantum entanglement of the data is measured, it will be safe from outside intervention." According to Einstein, Podolsky, and Rosen, each particle has an intrinsic state that totally controls the outcome of any given measurement." It is possible to secure the integrity of data already obtained from the quantum computer. Zero Knowledge can be used to test data at rest, and QKD combined with quantum authenticated encryption can be used to safeguard data in motion.

NTRUEncrypt is a public-key asymmetric encryption algorithm based on lattices. NTRUEncrypt is a lattice-based encryption system designed to guard against quantum computing that was initially introduced in 1996. "For comparison, we selected implementations with 128-bit security (with the exception of RSA-1024, which has 80-bit security). Even while RSA-1024 cannot match the same security level, our AvrNTRU surpasses the RSA implementation, decrypting 82.8 times faster ", The predecessor of NTRUPrime is NTRUEncrypt. NTRUPrime is a faster updated version of NTRUEncrypt that is currently in trials with the US government to see if it can be used to defend traditional computer systems against quantum computing. While quantum computing cannot produce enough qubits to pose a threat to today's encryption systems, van Oorschot-classical Wiener's computing parallel collision finding algorithms now pose a greater threat to integrity than quantum computing.

3.4 Non-Repudiation

Amiri and Andersson conducted an in-depth review of unconditionally secure quantum signatures in 2015, on which we heavily relied in setting the direction of our research into quantum non-repudiation. In their work, Amiri and Andersson describe signature schemes, whether for classical or quantum data, as having three goals: unforgeability, non-repudiation, and transferability. It should be impossible for an adversary to send a signed message impersonating a legitimate party, and impossible for a legitimate sender to deny a signed message. Furthermore, when one party verifies or rejects a signature they should be confident that any other party would verify or reject in the same way.

We chose to categorize the quantum signature algorithms we examined first according to whether they target classical or quantum data, as this is indicative of a major difference in their usage. Next, we examined whether the signature produced was reusable, i.e. could be verified

by more than one party without destroying the data. We also categorized the algorithms on the underlying security principle behind the quantum signature, as well as the role of a third party arbitrator in the protocol, if one was present at all.

Our selection of algorithms ranges from the original quantum signature scheme as proposed by Gottesman and Chuang in 2001 to recent research conducted by Hematpour, Ahadpour, and Behnia involving the dynamics of quantum dots, and covers a variety approaches to quantum non-repudiation. A summary of our reviewed protocols can be seen in

Common to most early schemes for quantum signatures, is the lack of a trusted third party, without which it is impossible to produce unconditionally secure signatures of quantum messages. In more recently developed quantum signature schemes, a trusted third party either generates private keys or provides non-repudiation. The only recent protocol we found which provides non-repudiation without the use of a trusted third party was proposed by Amerimehr and Dehkordi, and also provides confidentiality and integrity through the use of classical encryption and keyed hashing in combination with transmission over a quantum channel, though this protocol is not capable of signing quantum data.

CHAPTER 4

ANALYSIS & RESULTS

4.1 Implementation of a Quantum Cryptosystem

We decided it was vital to develop a theoretical algorithm presented in a research paper using a quantum programming environment because a key goal of this research was to identify quantum cryptography algorithms with practical applications. Because of its overall ease of use, quality of documentation, and convenience of usage when running against IBM's cloud-connected quantum hardware, Qiskit was chosen as our implementation tool of choice. We choose the algorithm proposed by Amerimehr and Dehkordi in their 2018 work "Quantum Symmetric Cryptosystem Based on Algebraic Codes" as the most promising candidate for implementation because it is simple and provides secrecy, integrity, and non-repudiation. For the sake of convenience, we'll refer to this cryptosystem as AD18, as BB84, B92, and others have done.

The algorithm is similar to BB84 in that the transmitting party, Alice, chooses random bases for her message transmission. While in BB84, the recipient, Bob, measures the received qubits on a random basis, in AD18, Bob measures all of the received message values at a 22.5^0 angle and encounters a 15% measurement error rate. The actual message sent by Alice contains an algebraic error-correcting code of sufficient quality to rectify the flaws Bob encounters, allowing for successful transmission without the need for public base notification. A keyed hash value is interspersed with the message qubits, and a pre-shared key is utilised in conjunction with some functions f(k) and g(k) to decide the places and bases used in the hash qubits transmission.

The quantum and classical components of the AD18 cryptosystem can be conceived of separately. Preparing, sending, and measuring qubits are all part of the quantum puzzle. Encryption, keyed hashing, and error correction are the traditional components of the algorithm.

We were able to use the BB84 code example from the Qiskit documentation to demonstrate the

preparation and measurement of qubits in various bases for the quantum component of the process. In BB84, this was achieved by combining negation (X) and Hadamard (H) gates during preparation and H gates during measurement. We used a R_z gate with theta=-π/8 in AD18 because Bob is measuring in a 22.5° basis.

Algorithm	**Target Data**	**Reusable?**	**Security Principle**	**Third Party**
Gottesman and Chuang 2001	Classical	Maybe	Quantum trapdoor function	None
Curty and Santos 2001	Classical	No	Entanglement	None
Barnum et al. 2002	Quantum	No	Purity testing codes	None
Kang et al. 2015	Quantum	No	Quantum trapdoor function	Provides non-repudiation
Amiri et al. 2016	Classical	Yes	Quantum key distribution	Provides non-repudiation
Chen et al. 2017	Quantum	No	One-Time pad	Arbitrator generates private keys
Amerimehr and Dehkordi 2018	Classical	Yes	Classical encryption and HMAC	None
Hematpour et al. 2020	Classical	No	Unique system state and critical points	Provides non-repudiation

Table 2. Classification of Quantum Non-Repudiation Algorithms

The traditional parts of the method, such as encryption and hashing, did not appear to be as significant as qubit measurement and error correction to the entire notion. For the encryption and decryption of the message, we chose a Salsa20 stream cypher with a 128-bit key, as well as an HMAC-MD5 keyed hash due to its short length. These protocols could readily be replaced with alternatives without affecting the algorithm's functionality, albeit the total number of bits conveyed and the protocol's overall security would be affected. We chose a simple implementation for our f(k) function, in which the message and hash are simply concatenated together, though this would need to be updated for a practical implementation to keep the

system secure. The g(k) function used to determine the keyed hash's transmission bases was similarly kept simple, with Alice sending hash bit H_i in B_z if $k_{(i \bmod len(k))} = 0$ and in B_x otherwise.

The identification of an algebraic error correcting code (ECC) capable of handling the observed error rates for non-trivial communications was the key challenge in effectively implementing this technique. The authors of the paper that describes AD18 send a two-bit message that their [5,2,3] linear error correcting code expands to five bits. We used the Python module commpy and employed its convolutional coding algorithms since we wanted to send the substantially lengthier message "hello world." Using a memory capacity of 3 and a G-Matrix initialised with the value array, a transmission success rate of >90% on a three character message was achieved ([[1, 3, 5, 7, 9, 11, 1, 3, 5, 7, 9, 11]]). Simpler convolutional code trellises significantly degraded the cryptosystem's performance, with transmission success rates of only <25% per byte for a value array ([[5,7]]). Furthermore, the G-Matrix employed in our implementation is unlikely to be practicable in a real-world implementation because it adds 100 bits of error correction each byte transmitted, which may not be realistic given current quantum channel bitrates.

The technique does not give a high level of consistency for the transmission of lengthier messages, which is a bigger challenge that is not unique to our implementation. An examination of the algorithm's performance over 1000 iterations using the aforementioned settings for string lengths ranging from 1 to 8 shows a success rate of roughly 0.96 per character communicated, thus we predict and observe a success rate of 0.96^8=0.72 for an 8 character message. While the error correction algorithms could be fine-tuned to improve performance, the probabilistic nature of quantum measurements combined with a hash check against the entire message means that failing to correctly decode even a single bit of data using ECC will result in a significantly different hash value and thus a failure to authenticate the message.

While we recognise that the authors are not experts in the subject of error correcting codes and that we may have utilised a suboptimal technique in our implementation, we still believe that the suggested approach is unworkable for non-trivial cases. Every bit measured in this cryptosystem has a 15% probability of being measured wrongly when Bob takes his

measurements. As a result, regardless of the ECC utilised, there is always a non-zero chance of a decoding error, and the overall error rate quickly approaches a prohibitive level. In a simple example with a two-bit message and three bits of error correction, the probability of Bob properly measuring four or more bits as necessary for error correction is $0.85^5+5*(0.15*0.85^4)=84\%$. If we use the same error correction mechanism and expand to 8 message bits with 12 bits of ECC, the chance of correctly measuring a sufficient number of bits to decode the message drops to 50%, and the chance of correctly measuring a sufficient number of bits to decode the message is cut in half with each subsequent 20-bit block of error corrected data we append.

4.2 General Process For Quantum Cryptographic Algorithm Implementation

We hope that discussing and generalising the methods employed in our implementation might aid other researchers looking to answer similar concerns about other quantum cryptography protocols, as the effort of building a suggested quantum algorithm can prove invaluable in discovering its flaws. Our implementation was done in Qiskit, but within the circuit-based quantum computing paradigm, this technique is language agnostic and should work with Cirq, Q#, AWS Braket, and other quantum programming languages. The approach provided here also has the drawback of not representing quantum data transmission, albeit this is mostly due to the current state of quantum programming.

The quantum cryptography algorithms we looked at are generally made up of the four steps and activities listed below:

1. Preparation: Establish pre-shared secrets, choose channels, and initialise values as qubits.
2. Sender Processing: Cryptographic operations on message qubits done by the sender. Data is transmitted across a specified channel, and public values are declared.
3. Recipient Processing: Message qubits are encrypted, data is verified, and eavesdropper detection is conducted.

As seen in multi-party signature schemes, when data is transmitted back and forth between the sender, recipient, and a trusted third party, with various operations applied along the route, steps can be repeated or reordered, and roles can change.

4.2.1 Preparation

In this stage of a cryptographic protocol, the sender and receiver establish a message channel, establish which pre-shared secret values to use, and prepare any necessary data. Should any classical cryptography, such as encryption or hashing, be required in the protocol, we recommend applying it at this stage whenever possible. Data preparation involving classical data can be done by converting into binary, then applying an X gate to qubits which are meant to represent 1s in the binary data. As classical data can be large, it can quickly push simulators to their limits when trying to perform even simple operations on a 32-bit or larger classical value. To avoid this pitfall, consider that it is typically not necessary to construct a complicated circuit in which all values are processed in parallel, unless all of the qubits interact in some manner. Instead, consider representing the interaction as an array of simpler circuits. For quantum data, in this stage the sender would apply appropriate operations in order to set the qubits into the appropriate state, such as executing Grover's algorithm up to the final measurement stage prior to hypothetically signing and transmitting the results of this algorithm.

We also recommend looking for places where initial preparation can be simplified without materially impacting the functionality of the protocol under study.

When implemented, the input to this stage should be classical data in the form of a string, byte array, bit array, or similar, and the output should be a quantum circuit or array of quantum circuits.

4.2.2 Sender Processing

This stage is the least specific as it will vary the most widely from protocol to protocol. The most common operations we observed in this stage were simple rotations, which are typically applied either as the Hadamard (H) gate or rotation ($R_{\{x,y,z\}}$) gates of arbitrary value. As the names of gates in quantum frameworks are generally well documented and only differ mildly amongst the various languages, referring to API documentation at this stage will resolve many challenges. Protocols which require custom unitary gate operations are also supported by many frameworks, such as Qiskit's UnitaryGate class.

When implemented, this stage will take as its input the quantum circuit that was produced in the preparation stage, and the output should be a quantum circuit or array of quantum circuits with

additional operations applied.

4.2.3 Transmission

The quantum computing frameworks we have used do not provide the capability to easily represent the transmission of quantum data from one party to another. If noise or an eavesdropper are to be present, they must be represented with additional gates and measurements at this stage. In a typical eavesdropper scenario, Eve makes measurements using any available public information, then would retransmit the data to Bob. In some cases Eve may apply her own set of gates before retransmitting the data.

This stage can be omitted in most cases and simply represented as the passing of the quantum circuit from the previous stage to the recipient processing stage. If an eavesdropper, noise, or other events which impact transmission are to occur in the simulation, then the stage should accept and return a quantum circuit.

4.2.4 Recipient Processing

This stage can be quite protocol dependent as well, though it will frequently involve the application of one or more gates prior to taking a final measurement of the received qubits. After measurement, additional public sharing or comparison of values may take place. This public sharing of values is limited by the same lack of capability described in the transmission stage, but is usually easily represented by passing parameters into a function. For signature or hashing schemes, this is where a final validation of the signature or hash will occur, either by computing a classical keyed hash or by performing additional quantum operations involving a third-party arbitrator.

The implementation of this stage should take a quantum circuit to represent the data received by the recipient, and return either a classical value as in the case of encryption and decryption, or a boolean value indicating the success of the signature or hash validation.

4.2.5 Workflow

Our proposed workflow follows an iterative model of development, similar to that seen in test-

driven development and agile methodologies. After identifying the desired algorithm for implementation, and validating that any quantum operations are supported by the chosen framework, the **first step** in this approach is to categorize the steps of the protocol into the previously described stages. In our implementations, we found it helpful to create a primary method to represent the full protocol, then create empty methods named for each stage, such as alice_prepare_message or bob_decrypt_message. For more complicated protocols, there will likely be multiple methods created for some of the stages. Before adding code to each method, it may be useful to describe the specific actions of the protocol in comments, and to identify which require classical data manipulation and which require quantum operations.

The **second step** of our workflow is to identify and initialize any values external to the algorithm under study, such as pre-shared secret keys or configuration of the quantum simulator. These should typically be defined as shared values outside the scope of any methods, as this makes them easier to locate and change and lowers the complexity of method signatures. We also highly recommend either using a standard logging library, or adding a boolean value debug = true so that helpful messages throughout the code can be conditionally enabled or disabled.

The **next step** in the workflow is to begin implementing the simplest possible version of the protocol by only coding the inputs and outputs of each stage. Use a small input of all zeros or a single letter string, create a quantum circuit with no gates or an identity gate for the transmission, only apply a measurement operation at the recipient, and perform any validation by simply returning true or false. This allows for easy verification that data is able to flow from end to end through the algorithm without introducing extra complexity. Judicious use of debugging statements at this stage can make later troubleshooting significantly simpler.

The **fourth step** should be to implement any classical operations used by the protocol, such as classical encryption, using appropriate libraries where possible. Validate that these operations function as expected outside of the quantum protocol under study, and add any relevant debugging statements.

In the **final step**, we implement the quantum portion of the algorithm within the structure we have created. Apply appropriate gate operations to the empty circuits that were constructed in the third step of the workflow, then test and verify the circuit on a local simulator. Once the

protocol is functioning as expected, perform any desired refactoring activities. Creating a test harness which performs a repeated execution of the protocol with a variety of inputs, and validates the outputs, is highly recommended for any subsequent analysis.

4.3 Example Using Kak's Three-Stage Protocol

Using the workflow described above, we present a simple implementation of Kak's Three-Stage Protocol for encryption of a single bit message. First, we categorize the steps of the protocol as follows:

• Preparation: Alice selects a value x to transmit. Alice and Bob each establish their own secret key.

• Sender processing 1: Alice applies $U_A = R(\theta)$ to her qubit.

• Transmission 1: Alice transmits to Bob

• Recipient processing 1: Bob applies $U_B = R(\phi)$ to the received qubit.

• Transmission 2: Bob transmits back to Alice

• Sender processing 2: Alice applies $U_A^\dagger$ to the qubit.

• Transmission 3: Alice transmits back to Bob

• Recipient processing 2: Bob applies $U_B^\dagger$ and has now decrypted the message.

As we are not planning to simulate an eavesdropper or noise in this protocol, all three transmission steps will not require any code and can simply be represented by comments and values being passed between methods. The stub of our Kak's protocol implementation after the completion of the first step of our implementation methodology can be seen in Listing.

Implementation of Kak's Three-Stage Protocol :

Step 1

```
def kak_3_stage(message):
    prepared = alice_prepare_message(message)
    transmission_1 = alice_apply_rotation(prepared)
    transmission_2 = bob_apply_rotation(transmission_1)
    transmission_3 = alice_remove_rotation(transmission_2)
    decrypted_message = bob_remove_rotation(transmission_3)
    return decrypted_message
```

In the second step of our methodology, we can see that Kak's protocol uses two secret keys,

which we can initialize as desired. As the secret keys are used to generate the θ and ϕ values used by Alice and Bob, we choose to generate a random value for each between 0 and 2π in order to keep the implementation simple. As seen in Listing, we also create a debug flag and define **backend = qasm_simulator** for use later.

Step 2

```
debug = True      # Change to False to remove messages
backend = 'qasm_simulator'
alice_key = np.random.uniform(0, 2*pi)
bob_key = np.random.uniform(0, 2*pi)
if debug:
    print("Alice's key: %s" % alice_key)
    print("Bob's key: %s" % bob_key)
```

Next, we can implement the simplest version of our methods in which Alice and Bob simply apply an I gate at each step, and Bob performs a measurement in the final step. While most of these methods are trivial, the final one in which Bob performs his measurement will contain the code necessary to execute the quantum circuit on a simulator. The method for Bob's final rotation removal and measurement appears in Listing 3

Step 3

```
def bob_remove_rotation(qc):
    qc.i(0)
    qc.measure(0,0)
    if debug:
        print('After Bob removes rotation and measures')
        print(qc)
    # Execute in simulator
    qasm_sim = Aer.get_backend(backend)
    qobj = assemble(qc, shots=1, memory=True)
    result = qasm_sim.run(qobj).result()
    measured_bit = int(result.get_memory()[0])

    return measured_bit
```

As the protocol contains no classical functions, we move to the final step where we properly

prepare Alice's message and implement rotation operators to replace the identity gates we used in the previous step. As Qiskit provides an R_Z gate capable of performing an arbitrary rotation, we can use Alice and Bob's secret key values directly in this gate to apply their rotations. In the code, this simply involves replacing **qc.i(0) with qc.rz(alice_key, 0),** as seen in Listing. After completing the implementation, we can verify that the operation works correctly by checking **kak_3_stage(0) == 0 and kak_3_stage(1) == 1.** From this starting point, it would be possible to refactor the protocol to take a longer input, use a more complicated key generation method, or include an eavesdropper.

Step 5

```
def alice_remove_rotation(qc):
    qc.rz(-alice_key, 0)
    if debug: print(qc)
    return qc
```

4.3 Discussion

4.3.1 Confidentiality

While quantum computing's contributions to confidential communications are most immediately going to come by way of Quantum Key Distribution paired with classical cryptography, it is clear that it could play a role in communication itself as the capabilities of quantum computers and networks improve.

Kak's Three-Stage Protocol is a foundational demonstration of entirely quantum encrypted communications. The protocol is capable of perfect security when accompanied by a classical protocol to ensure identity of communicating parties and a mechanism for error-checking/-correction. Kak's Three-Stage Protocol has been extended by further research that address its shortcomings, including implementations that enable multi-photon transmissions and the correction of errors that occur over a quantum channel. While these advances bring the realization of an entirely quantum cryptosystem closer to near-term implementation in real-world applications, such solutions are limited by the capacity of circuit-based quantum processors which, at the time of writing, operate with less than 100 qubits.

The scheme proposed by Amerimehr and Dehkordi ensures confidentiality by using a single pre-shared encryption key, which is an advantage over similar schemes that rely on multiple

secret keys as a single key reduces pre-processing overhead. In addition to providing a performance benefit, this pre-shared key can be reused securely, which is a desirable characteristic of a quantum cryptosystem that enhances its real-world viability. While this system's theoretical capabilities are promising for real-world implementation, its reliability quickly degrades as message length increases, as demonstrated in 4.1. This is, again, largely due to the limited capacity of quantum computers and suggests that the cryptosystem should not be considered for real-world applications until the capacity of quantum processors increases or a more efficient algebraic ECC is implemented.

Despite their additional computational overhead, hybrid quantum cryptosystems like the one proposed by Pleșa are somewhat less reliant on the advancement of quantum computers because they transmit encrypted messages over classical channels. As a result, these systems have a greater likelihood of real-world implementation in the near future. Pleşa's system achieves confidentiality by use of a quantum circuit that guarantees perfect randomization. While its ability to ensure confidentiality can be proven through experimentation, its real-world application is hindered by the requirement for each communicating party to share a pair of entangled qubits. The challenges introduced in maintaining and transporting entangled qubits are not unique to this scheme but are notable obstacles that are likely to delay the real-world implementation of this system.

4.3.2 Non-Repudiation

Quantum non-repudiation is challenging due to the the impossibility of unconditionally secure signatures for quantum data without a trusted third party, as well as the destructive nature of measurements for both signatures and data. Quantum signatures are frequently not reusable, which makes their transferability questionable. Quantum signatures of classical data are unlikely to be very important in the near future as classical signatures are still viable, and NIST is actively researching post-quantum signature protocols for classical data.

Quantum signatures of quantum data will become more important in the future when quantum computers become more prevalent. Just like current certificates use trusted root authorities, quantum signatures will need trusted quantum arbitrators, and this concept has not been well

explored to date. Several promising quantum signature schemes have been experimentally realized in recent years, including an implementation by An et al. in 2019, and another by Ding et al. in 2020. Both of these practical implementations are based off of a proposal by Amiri et al.in 2016 for a secure quantum signature scheme which functions over insecure channels.

In the signature protocol proposed by Amiri et al., the sender, Alice, communicates with two receiving parties, Bob and Charlie. Alice generates correlated bit strings separately with Bob and Charlie using a key generation protocol such as BB84. Bob and Charlie exchange half of their generated bit strings with one another over a secure classical channel. Alice signs and transmits a message over a classical channel to her desired recipient, say Bob, who then checks the signature against his own key and the key received from Charlie. If the number of mismatches between the signature and secret key is below some limit, then Bob can accept the message. Bob is also able to transmit the message and signature to Charlie for verification in the same manner. As quantum channels are only required during the key generation stage of this protocol, it is well suited for implementation as BB84 or other QKD protocols can be used

CHAPTER 5

CONCLUSION

In this paper we have explored the impacts of quantum computing on classical cryptography, examined the current state of the art of quantum cryptography, implemented and analyzed Amerimehr and Dehkordi's 2018 symmetric quantum encryption algorithm, and presented a generalized process for the implementation of quantum cryptographic algorithms. The impacts of quantum computing on today's cryptographic systems are well understood, and efforts are being made by NIST to select suitable replacements for vulnerable public-key encryption, key-establishment, and signature algorithms. We identified promising quantum algorithms for important aspects of cryptography, including twin-field QKD for key distribution, zero knowledge proofs for integrity, AD18 for encryption, and Amiri et al. for non-repudiation. While some of these algorithms, such as AD18, appear to need additional research and refinement to be practical for many applications, they can serve as a basis from which to build for future research.

To aid in future evaluations of quantum cryptographic protocols, we examined implementations of BB84, AD18, and Kak's Three-Stage Protocol, and proposed a standard process by which other researchers can approach the challenge of implementing quantum cryptographic algorithms. As the field of quantum cryptography is still in its infancy, our hope is that these final two contributions will be of particular value. Finally, we will provide recommended actions organizations can take today and in the near future to prepare themselves for advances in quantum computing and cryptography.

5.1 Recommendations for Today

Our first recommendation for organizations today is to gain an awareness of quantum computing, what problems it can and cannot be used to solve, and how it threatens current cryptographic systems. Outside of cutting edge security or research concerns, we do not believe it is necessary to immediately hire quantum specialists, but having an awareness of the state of the art in quantum computing will likely provide a competitive advantage to businesses over the next few years. Software engineers and security experts would benefit from learning

the basic concepts of quantum computing as well, just as they have been encouraged to do with concepts such as cloud computing and machine learning in the recent past.

As today's most commonly used encryption systems are either weakened or broken by quantum computing, we highly recommend that companies be prepared to implement post-quantum encryption once final candidate algorithms are approved by NIST. Though quantum computers will not be capable of breaking RSA2048 for many years, encrypted data could still be captured by an adversary today and stored until decryption becomes feasible in the future. For data at rest, we recommend using AES with a 256-bit key length, and discontinuing the usage of 3DES and AES with 128- or 192-bit key lengths, as the impact of Grover's algorithm lowers the effective key lengths below those recommended by NIST in Section 3.4 of Special Publication 800-175B.

5.2 Recommendations for the Future

It is extraordinarily difficult for experts to make accurate predictions of future developments in any field. The authors of this work make no claims to be more than novices in the field of quantum computing, but we felt we would be remiss to not provide a couple of broad predictions and accompanying recommendations for the next ten to twenty years.

In the next two decades, we predict that quantum computing will become widespread. Quantum computers will become more capable and easier to access, and an increasing number of companies will hire quantum computing experts. Quantum algorithms will be commonly used for applications such as entropy generation, key distribution, and optimization problems. If dramatic hardware advancements are made, we may even see specialized quantum processing units with a small number of qubits appear in personal computers. To stay ahead of these predictions, we reiterate our previous recommendation that organizations begin building their quantum computing capabilities. We also recommend that organizations be wary of the inevitable wave of charlatan companies that will promise expensive quantum computing offerings which are capable of solving all the world's problems.

As quantum computers become ubiquitous and capable, we predict that large numbers of new algorithms will be discovered. These algorithms will have broad impacts, ranging from threatening the security of previously safe cryptographic protocols, to producing rapid advances in materials science, medicine, and artificial intelligence. The technological and

sociopolitical impacts of these new discoveries may alter society as fundamentally as industrialization, automobiles, aviation, and the internet did. Our recommendation is that organizations be prepared to adapt to rapid paradigm shifts in the security and technology landscape.

5.3 Future Research

In future work, we would like to further research and explore the encryption of quantum data, as this part of the field appears to be underdeveloped when compared to other aspects of quantum cryptography. In particular, we would like to research whether encryption protocols could diverge to handle quantum data at rest and in motion, as we have seen symmetric and asymmetric encryption develop in the classical realm. We believe an opportunity exists to further generalize our process for implementing quantum cryptographic algorithms, so that it can apply to non-circuit paradigms of quantum computing. We would also like to further explore the space of algebraic error correcting codes in order to revisit AD18 and improve the efficiency and overall capability of the protocol.

REFERENCES

Quantum Intro:

[1] Rieffel, E., & Polak, W. (2000). An introduction to quantum computing for non-physicists. ACM Computing Surveys (CSUR), 32(3), 300-335.

[2] Humble, T. (2018). Consumer applications of quantum computing: A promising approach for secure computation, trusted data storage, and efficient applications. IEEE Consumer Electronics Magazine, 7(6), 8-14. 54, no. 2, pp. 614–629, Feb. 2015, doi: 10.1007/s10773-014-2254-y.

[3] Albash, T., & Lidar, D. A. (2018). Adiabatic quantum computation. Reviews of Modern Physics, 90(1), 015002.

[4] Bauckhage, C., Brito, E., Cvejoski, K., Ojeda, C., Sifa, R., & Wrobel, S. (2017). Adiabatic Quantum Computing for Binary Clustering. arXiv preprint arXiv:1706.05528.

[5] Ying, M. (2010). Quantum computation, quantum theory and AI. Artificial Intelligence, 174(2), 162-176.

[6] Gisin, N., Ribordy, G., Tittel, W., & Zbinden, H. (2002). Quantum cryptography. Reviews of modern physics, 74(1), 145.

[7] Ablayev, Farid et al. "Quantum Fingerprinting and Quantum Hashing. Computational and Cryptographical Aspects." Baltic Journal of Modern Computing 4.4 (2016): 860–. Web.

Quantum Encryption:

[1] A. Amerimehr and M. H. Dehkordi, "Quantum Symmetric Cryptosystem Based on Algebraic Codes," IEEE Communications Letters, vol. 22, no. 9, pp. 1746–1749, Sep. 2018, doi: 10.1109/LCOMM.2018.2844245.

Quantum Key Distribution:

[1] A. I. Nurhadi and N. R. Syambas, "Quantum Key Distribution (QKD) Protocols: A Survey," 2018 4th International Conference on Wireless and Telematics (ICWT), Nusa Dua, 2018, pp. 1-5, doi: 10.1109/ICWT.2018.8527822.

[2] X. Liu et al., "Multi-path based Quasi-real-time Quantum Key Distribution in Software Defined Quantum Key Distribution Networks (SD-QKDN)," 2019 18th International Conference on Optical Communications andNetworks (ICOCN), Huangshan, China, 2019, pp. 1-3, doi: 10.1109/ICOCN.2019.8934684.

[3] W. Yu, Y. Zhou, X. Zhou, L. Wang and S. Chen, "Study on StatisticalAnalysis Method of Decoy-state Quantum Key Distribution with Finite-length Data," 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chongqing, China, 2020, pp. 2435-2440, doi: 10.1109/ITNEC48623.2020.9084715.

[4] C. H. Park, M. Ki Woo, B. K. Park, Y. -S. Kim, S. Kim and S. -W. Han, "Research on Plug-and-Play Twin-Field Quantum Key Distribution," 2020International Conference on Information and Communication Technology Convergence (ICTC), Jeju Island, Korea (South), 2020, pp. 890-893, doi: 10.1109/ICTC49870.2020.9289265.

Quantum Non-Repudiation:

[1] N. Hematpour, S. Ahadpour, and S. Behnia, “Presence of dynamics of quantum dots in the digital signature using DNA alphabet and chaotic S-box,” Multimed Tools Appl, Nov. 2020, doi: 10.1007/s11042-020-10059-5. [Online]. Available: https://doi.org/10.1007/s11042-020-10059-5.

[2] M.-S. Kang, C.-H. Hong, J. Heo, J.-I. Lim, and H.-J. Yang, “Quantum Signature Scheme Using a Single Qubit Rotation Operator,” Int J Theor Phys, vol. 54, no. 2, pp. 614–629, Feb. 2015, doi: 10.1007/s10773-014-2254-y.

[3] Georgios M. Nikolopoulos, and Marc Fischlin. “Information-Theoretically Secure Data Origin Authentication with Quantum and Classical Resources.” Cryptography 4.4 (2020): 31–. Web.

[4] M. Curty and D. J. Santos, “Quantum authentication of classical messages,” Phys. Rev. A, vol. 64, no. 6, p. 062309, Nov. 2001, doi: 10.1103/PhysRevA.64.062309.

Integrity and Post-Quantum Security:

[1] A. Chailloux, M. Naya-Plasencia, and A. Schrottenloher, “An Efficient Quantum Collision Search Algorithm and Implications on Symmetric Cryptography,” in Advances in Cryptology – ASIACRYPT 2017, Cham, 2017, pp. 211–240, doi: 10.1007/978-3-319-70697-9_8.

Research Methodology:

[1] N. Jorstad and L. T. Smith, “Cryptographic Algorithm Metrics,” in Proceedings of the 20th National Information Systems Security Conference, Baltimore, Maryland, United States, 1997 [Online]. Available: https://csrc.nist.gov/csrc/media/publications/conference-paper/1997/10/10/proceedings-of-the-20th-nissc-1997/documents/128.pdf

[2] E. Khan, S. Meraj, and M. M. Khan, “Security Analysis of QKD Protocols: Simulation Comparison,” in 2020 17th International Bhurban Conference on Applied Sciences and Technology (IBCAST), 2020, pp. 383–388, doi: 10.1109/IBCAST47879.2020.9044522.