Quantum Key Distribution Simulation using Entangled Bell States

A Senior Project
presented to
the Faculty of the Physics Department
California Polytechnic State University San Luis Obispo

In Partial Fulfillment
of the Requirements for the Degree
Bachelor of Science

by
Nayana Tiwari
June 2022

Abstract:

To communicate information securely, the sender and recipient of the information need to have a shared, secret key. Quantum key distribution (QKD) is a proposed method for this and takes advantage of the laws of quantum mechanics. The users, Alice and Bob, exchange quantum information in the form of entangled qubits over a quantum channel as well as exchanging measurement information over a classical channel. A successful QKD algorithm will ensure that when an eavesdropper has access to both the quantum and classical information channels, they cannot deduce the key, and they will be detected by the key generators. This paper will introduce quantum key distribution and explain the implemented simulation of a proposed QKD algorithm using entangled Bell states. The proposed T22 protocol was compared against the more common BB84 QKD protocol. The results show that it takes 3x longer to generate a key of length *m* bits using the T22 protocol, however the T22 protocol is 36x more secure than BB84.

Table of Contents

Table of Tables

Table of Figures

I. Introduction

Classical encryption requires key establishment using complex algorithms such as RSA encryption, however, the mathematics behind the encryption algorithms are still vulnerable. Quantum key distribution (QKD) allows two users with separate quantum computers to create an encryption key. The users communicate over the Internet (classical channel) and have a method of exchanging qubits between each other (quantum channel). If an eavesdropper is snooping on either or both the classical and the quantum channel, they should not be able to decode the encryption key, and the users should be able to detect an eavesdropper. Once an encryption key has been generated, the users can communicate over a classical channel securely. This process is known as quantum key distribution and is a field of quantum cryptography.

Several quantum key distribution algorithms have been proposed, the first being Bennet and Brassard's BB84 protocol in 1984 followed by Eckert's E91 protocol published in 1991 which was based on entanglement [1]. The protocol in this paper uses entangled Bell states to generate a quantum key and is based on a theoretical proposition by Dan Song and Dongxu Chen [2].

In this paper, section II details the theoretical background of quantum computing, section III presents the T22 QKD protocol, section IV shares the results, section V discusses the conclusions, section VI gives acknowledgements, and section VII includes the references.

II. Theoretical Background

i. Qubits

Quantum computing relies on the creation and manipulation of quantum bits (qubits). Qubits can be physically represented in the form of photons, atoms, and electrons using specified states to serve as a “0” and a “1”. An example is using the polarization of a photon with “0” being linearly polarized to 0° and “1” being linearly polarized to 90°. The state of a qubit must be able to be initialized, altered, and measured. The qubit collapses from superposition of “0” and “1” to one of the two states when measured unlike a classical bit which has a constantly known state of a “0” or a “1”. Quantum computing relies on Dirac notation to describe the “0” and “1” states. A single qubit state or wavefunction can be written as the following where α and β are complex, normalized values meaning $|\alpha|^2 + |\beta|^2 = 1$. This normalization ensures that the probability to be in any of the possible states is always 1.

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle \qquad (1)$$

ii. Single Qubit Gates

The operations that are performed on qubits are referred to as gates similar to classical computing. Gates can be represented using operators or matrices. The X or NOT gate flips the coefficients in the wavefunction as shown below acting on Equation 1. Note that the X gate is the same as the σ_x Pauli spin matrix.

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \qquad X|\Psi\rangle = \beta|0\rangle + \alpha|1\rangle$$

The Z gate negates β when acting on Equation 1 as show below. Note that the Z gate is the same as the σ_z Pauli spin matrix.

$$Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad Z|\Psi\rangle = \alpha|0\rangle - \beta|1\rangle$$

The states $|0\rangle$ and $|1\rangle$ are associated with the Z basis of measurement. However, other measurement bases can be used. For example, the BB84 protocol relies on also measuring in the X basis. The Hadamard gate transforms a qubit in the Z-basis in state $|0\rangle$ or $|1\rangle$ to the X-basis states $|+\rangle$ and $|-\rangle$ respectively. The X-basis is a rotation of the Z-basis by 90°. Below is the Hadamard gate acting on Equation 1.

$$H = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad H|\Psi\rangle = \frac{(\alpha+\beta)|0\rangle+(\alpha-\beta)|1\rangle}{2}$$

A qubit in the $|0\rangle$ state that is measured in the Z-basis has a 100% chance of being measured as a $|0\rangle$. Note that the coefficients in front of each state is known as the probability amplitude. The probability of measuring a $|0\rangle$ in Equation 1 is $|\alpha|^2$ and the probability of measuring a $|1\rangle$ is $|\beta|^2$. If a qubit is in the Z-basis state of $|0\rangle$ or $|1\rangle$ but is then measured in the X-basis, the measurement has an equal likelihood of being $|+\rangle$ or $|-\ \rangle$ due to the orthogonal relationship between the bases. The same outcome of an equal superposition is observed for a qubit that is in the X-basis states of $|+\rangle$ or $|-\rangle$ where the outcome is an equal likelihood between $|0\rangle$ and $|1\rangle$ when measured in the Z-basis.

iii. Two Qubit Gates

Two qubits can be entangled meaning the operations and measurements performed on one qubit affect both qubits. An example of a two qubit wavefunction is shown in Equation 2 where $|00\rangle$ means both qubits are in state $|0\rangle$, $|01\rangle$ means the first qubit is in state $|0\rangle$ and the second qubit is in state $|1\rangle$ and so forth.

$$|\Psi\rangle = \alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \delta|11\rangle \tag{2}$$

A two-qubit system is entangled if its wavefunction cannot be factored out into a wavefunction of the first qubit times the wavefunction of the second qubit. Examples of an entangled state are $|\Psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ or $|\Psi\rangle = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle - |11\rangle)$.

To entangle two qubits an entangling two-qubit gate must be used. A common entangling two qubit gate is known as the controlled NOT (CNOT) or controlled X (CX) gate. The CNOT gate has an enable or control qubit which performs an X or NOT gate on the target qubit if the enable qubit is in state $|1\rangle$. Similar to a classical enable, there is no effect on either qubit if the enable qubit is in state $|0\rangle$. As shown below, the CNOT gate acting on a two qubit state such as Equation 2 shall result in the swapping of the coefficients of $|10\rangle$ and $|11\rangle$. Depending on the values of the coefficients, the CNOT gate can entangle the two qubits.

$$CNOT = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \qquad CNOT|\Psi\rangle = \alpha|00\rangle + \beta|01\rangle + \delta|10\rangle + \gamma|11\rangle$$

A quantum circuit is when many gates operate on one or multiple qubist. Quantum gates must be unitary meaning, in the matrix form, the Hermetian adjoint (conjugate transpose) is the same as its inverse. Due to this property, operations can be reversed back to the initial state by performing the same gates in the opposite order. This principle is known as reversibility.

The maximum entanglement is limited by Bell's inequality and the two qubit states that are maximally entangled are referred to as Bell states [2]. As a reference, the resulting Bell state wavefunctions and circuits as performed on two qubits in the $|00\rangle$ state are shown in Figure 1. Note that the single qubit X, Z, and Hadamard gates are represented by boxes with an X, Z, and H respectively. The CNOT gate is represented with a solid dot on the enable qubit and a plus sign on the target qubit.

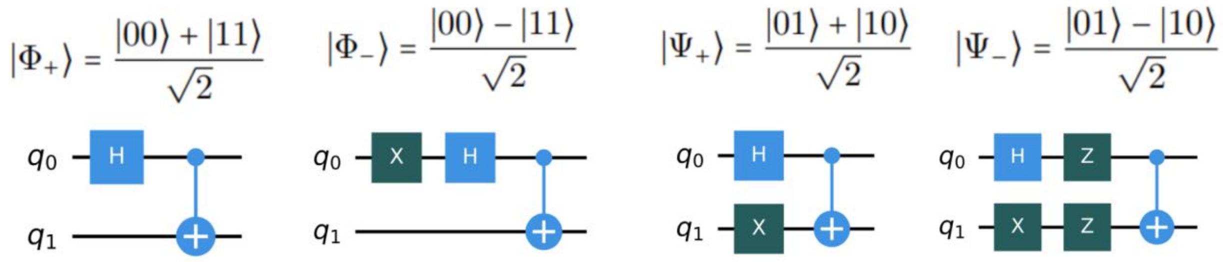
$|\Phi_+\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}}$
q_0
H
q_1
$|\Phi_-\rangle = \frac{|00\rangle - |11\rangle}{\sqrt{2}}$
q_0
X
H
q_1
$|\Psi_+\rangle = \frac{|01\rangle + |10\rangle}{\sqrt{2}}$
q_0
H
q_1
X
$|\Psi_-\rangle = \frac{|01\rangle - |10\rangle}{\sqrt{2}}$
q_0
H
Z
q_1
X
Z

Figure 1. The four Bell states represented by a wavefunction and circuit generated in Qiskit.

iv. Quantum Key Distribution

The two users described earlier who are involved in quantum key distribution are given the names Alice and Bob. Alice prepares several qubits, sends them to Bob, and Bob measures the qubits. Alice and Bob exchange some information over the classical channel to ensure Bob measured correctly. Then Alice and Bob confirm all the choices made for a few of the circuits to confirm there is no eavesdropper, Eve, on the line. The operations Alice and Bob perform as well as the information they share is dependent on the quantum key distribution protocol used.

The BB84 protocol uses single qubits that can be in either a $|0\rangle$ or a $|1\rangle$ state in the Z-basis or in the $|+\rangle$ or $|-\rangle$ state in the X-basis. Alice initializes qubits randomly between the four states then sends them to Bob over the quantum channel. Bob measures the qubits in either the Z or X basis. Bob shares the bases he measured in with Alice over classical channels and Alice confirms whether she measured the same. Alice and Bob also confirm some measurements to determine whether there was an eavesdropper on the line. Then they use the initialization or measurement of either a 0 or a 1 state in the respective basis as a bit in the key. For example, if Alice prepared a qubit in the $|+\rangle$ state and Bob measured in the X-basis, a “0” would be added to the key. [3] Alice chooses to measure between 2 bases so she has a ½ chance of picking the same basis as Bob – either they both pick the Z-basis or they both pick the X-basis. In BB84 the probability of generating a correct circuit (both Alice and Bob measure in the same basis) is:

$$P_{correct} = (Alice_{basis}) \times (Bob_{same\ basis})$$

$$P_{correct} = \frac{1}{2} \times 1 = \frac{1}{2}$$

Also note that for Eve to successfully infiltrate a circuit, she must guess the same circuit as Alice. Since there are two choices of basis for Eve, this means the probability of Eve infiltrating a circuit without being detected is:

$$P_{infiltrate} = (Alice_{basis}) \times (Eve_{basis})$$

$$P_{infiltrate} = \frac{1}{2} \times \frac{1}{2} = \frac{1}{4}$$

The BB84 protocol will be compared against the T22 protocol described in the next section.

III. T22 protocol

The new protocol is based on the Bell states described in the previous section and Song and Chen's "Quantum Key Distribution Based on Random Grouping Bell State Measurement" [2] which described a proposed QKD protocol using quantum computers and Bell state measurements. For this paper, the new protocol will be referred to as "T22" to follow convention of using the last initial and year of creation. Bell state measurements (similar to X and Z basis measurements) can distinguish between states in the Bell basis: $|\Phi_+\rangle, |\Phi_-\rangle, |\Psi_+\rangle, |\Psi_-\rangle$. In this protocol, Alice and Bob can choose between pairings and groupings. Each circuit consists of four qubits and as with any four objects, they can be paired in 3 different ways as represented by Table 1 with labels "A", "B", and "C" for ease of reading. The pairings decide how the four qubits will be entangled using CNOT gates.

Table 1. Possible pairings of four qubits (q0, q1, q2, q3) with labels for continued use in this paper.

Pair 1	Pair 2	Label
(q0, q1)	(q2, q3)	A
(q0, q2)	(q1, q3)	B
(q0, q3)	(q1, q2)	C

Alice and Bob also choose groupings which indicate which Bell state operations they use on the qubit pairings as shown in Table 2. Note that the groupings are binary numbers 0 to 3 and become the bits of the encryption key. The groupings determine the circuits from Figure 1 that are applied to each pair to entangle them.

Table 2. Possible Bell state groupings for the pairs of qubits. Each pair will now be entangled.

Pair 1	Pair 2	Groupings
$\|\Phi_{+}\rangle$	$\|\Phi_{-}\rangle$	00
$\|\Phi_{-}\rangle$	$\|\Phi_{+}\rangle$	01
$\|\Psi_{+}\rangle$	$\|\Psi_{-}\rangle$	10
$\|\Psi_{-}\rangle$	$\|\Psi_{+}\rangle$	11

In the T22 protocol, Alice randomly chooses both a grouping and a pairing and performs those operations on a four qubit circuit. Instead of measuring in the Bell basis as proposed in [2], Bob chooses a random grouping and pairing and performs the reverse operations. If Alice and Bob chose the same groupings and pairings, they will have created a mirrored circuit as shown in Figure 2. Considering the reversibility of quantum circuits, Bob will always measure the initial condition or all 0s, $|0000\rangle$.

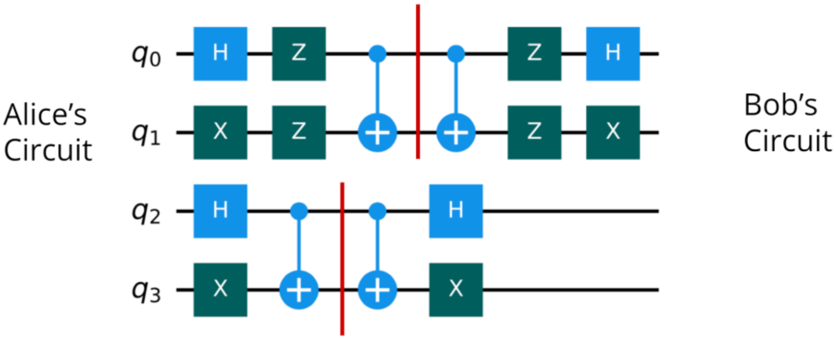

Figure 2. A mirrored circuit that will measure all 0s. Alice and Bob chose pairing 'A' and grouping '11'.

If Alice and Bob do not choose the same pairings and groupings, the qubits will be in a superposition of 1s and 0s at the end. Bob could measure many different states such as $|0010\rangle, |1011\rangle, |1111\rangle$, or even $|0000\rangle$ with varying probabilities. An example of the same pairings and different groupings is shown in Figure 3; an example of different pairings and the same groupings is shown in Figure 4, After Alice and Bob generate a circuit, Bob makes a measurement in the Z basis. He will measure one of the 16 possible states.

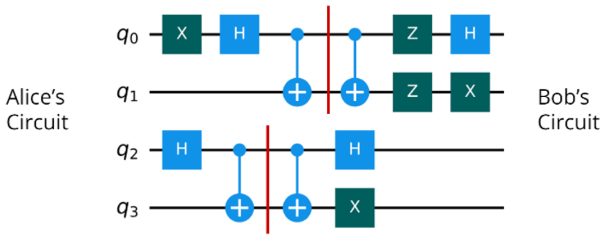

Figure 3. Alice and Bob choose pairing A, Alice chooses grouping 01, Bob chooses grouping 11.

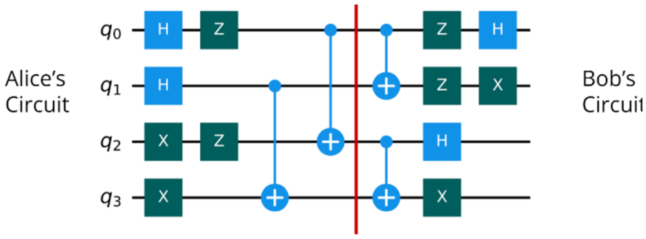

Figure 4. Alice chooses pairing B, Bob chooses pairing A, Alice and Bob choose grouping 11.

The final modification made to take advantage of the software simulation was to run each generated circuit for 256 shots. If Alice and Bob have the same pairings and groupings, all 256 shots should result in a $|0000\rangle$ measurement. If they do not have mirrored circuits, the 256 shots will result in a distribution of measurements across many different four qubit states. Thus, Bob, in the software simulation, knows he correctly chose his groupings and pairings if the 256 shots all return with the measurement $|0000\rangle$. Bob tells Alice over the classical channel the circuits they got correct without sharing pairing and grouping information. They also verify a couple circuits in their entirety by sharing the pairings and groupings used to check for an eavesdropper and verify a secure line. Finally, they use the leftover groupings as the key. An example of the T22 protocol is shown in Table 3.

Table 3. T22 QKD protocol example using 7 4-qubit circuits to generate a 6 bit key "010001."

Protocol Steps	Circuit 1	Circuit 2	Circuit 3	Circuit 4	Circuit 5	Circuit 6	Circuit 7
Alice pairing	A	B	C	A	B	C	A
Alice group	11	10	01	00	11	10	01
Bob pairing	B	B	C	A	B	C	A
Bob group	11	10	01	00	00	10	01
Eve detection		Verified				Verified	
Final Key			01	00			01

In the example, Bob and Alice choose pairings and groupings but circuits 1 and 5 will not result in a correct measurement so they are discarded. Then, circuits 2 and 6 are used to verify that there is no eavesdropper on the line. Thus, the groupings of circuits 3, 4, and 7 are used to generate the final key which is "010001."

Alice and Bob must choose the same pairings and groupings to result in a correct measurement in this protocol. Thus, using the T22 protocol, the probability of generating a correct circuit is:

$$P_{correct} = \left(Alice_{pair}\right) \times \left(Alice_{group}\right) \times \left(Bob_{same\ pair}\right) \times (Bob_{same\ group})$$

$$P_{correct} = \frac{1}{3} \times \frac{1}{4} \times 1 \times 1 = \frac{1}{12}$$

Note that the probability of generating a correct circuit using the T22 protocol is 6x less than the BB84 protocol. This can be an advantage since it is much harder for an eavesdropper to successfully infiltrate the quantum key generation. Any modification by an eavesdropper will result in Bob not measuring $|0000\rangle$ for all shots.

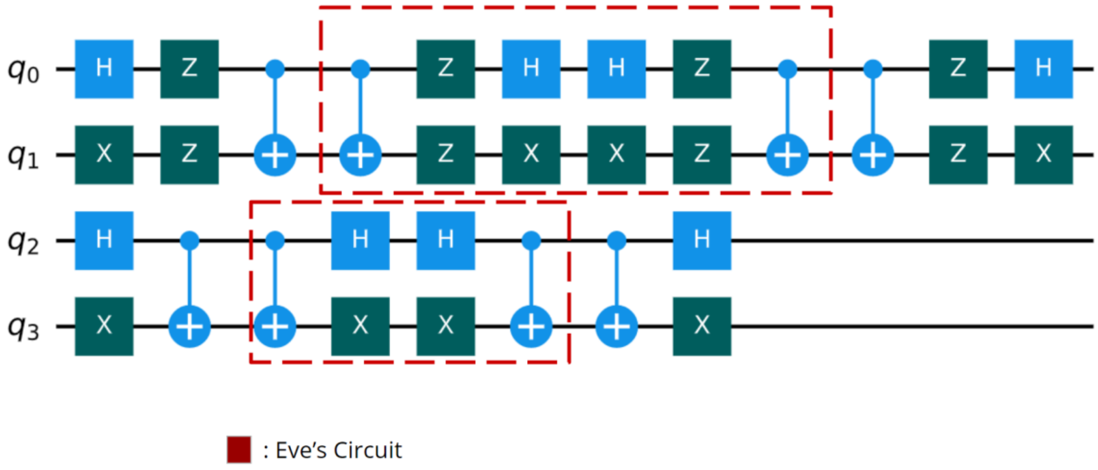

Figure 5. Successful eavesdropping for T22 QKD protocol using pairing A and grouping 11

An example of a successful eavesdropping is shown in Figure 5. Eve must reverse Alices circuit, measure all 0s, then again perform the original operations so Bob cannot detect Eve. This means Eve would have to guess the exact pairings and groupings that Alice chose every circuit to ensure that she remains undetected during the verification step of the protocol. The probability of Eve infiltrating one circuit successfully by guessing Alice's choices correctly is:

$$P_{infiltrate} = \left(Alice_{pair}\right) \times \left(Alice_{group}\right) \times \left(Eve_{pair}\right) \times \left(Eve_{group}\right)$$

$$P_{infiltrate} = \frac{1}{3} \times \frac{1}{4} \times \frac{1}{3} \times \frac{1}{4} = \frac{1}{144}$$

The BB84 protocol was similarly modified to use 256 shots. In the simulation, Bob and Alice do not need to share the measurement bases since Bob knows which circuits were correct based on a $|0\rangle$ measurement for all shots. Instead, they use the measurement bases as the bits for the key where Z-basis is a "0" and the X-basis is a "1." An important difference to note is the BB84 uses 1 qubit per circuit to generate 1 bit for the key while the T22 protocol uses 4 qubits per circuit to generate a 2 qubit key.

The QKD model of Alice, Bob, and Eve was modified to accommodate a software simulation of QKD rather than individuals with two quantum computers and a quantum channel. In the simulation, users are represented by JSON files to keep the flow of information the same – for example, the JSON file representing Alice will never contain Bob's measurements. Additionally, the online circuit simulators such as Qiskit cannot directly perform Bell state measurements so both protocols were modified to account for only measuring in the Z basis for consistency. Note that an X-basis measurement can be modified to the Z-basis using a Hadamard gate.

IV. Results

Both the modified BB84 protocol and the T22 protocol were run for 1000 samples using the QuTech Quantum Inspire software simulator backend and the Qiskit API for Python. The Quantum Inspire compiler performed optimizations on the circuits. The test fixed the random seed for Alice and the random seed for Bob for both protocols using two different system times in the execution of the test program. The code repository can be found on Github [4]. The results of the test are shown in Table 4.

Table 4. Software simulation results for 1000 circuits of each protocol.

Protocol	Num. of Circuits	Num. of Successes	Predicted Num. of Successes	Key Length (bits)
Modified BB84	1000	490	500	490
T22 protocol	1000	85	83.33	166

The predicted number of successes closely matched the actual number of successful circuits suggesting proper implementation of protocols. Note that the probability of a correct T22 circuit is 1/12 however 2 bits of the key are generated per circuit while the probability of a correct BB84 circuit is 1/2 with 1 bit generated per circuit. Thus, to generate a key of length *m* the T22 protocol should be run 6*m* times while the BB84 protocol should be run 2*m* times (not including verification circuits). Thus, the resulting key length of the T22 protocol is about 1/3 the length of the BB84 protocol which is expected. Although using the T22 protocol will take 3x longer to generate a key of the same length as the BB84 protocol, the odds of Eve successfully infiltrating a circuit is 36x less than BB84.

V. Conclusion

The software simulation confirmed the theoretical expectations for the number of successful circuits and showed that the T22 protocol still generated a key of a third the length of BB84 with the added security. The T22 protocol is much more secure than BB84 since the eavesdropper is 36x less likely to guess a circuit correctly. Note that 3x the number of circuits with 4x the number of qubits per circuit for the T22 protocol. However, the time needed to run additional and larger circuits to generate a key of the desired length could be outweighed by the added security benefits. The feasibility of the T22 protocol depends on the capabilities of actual quantum computers – whether the computer can run four qubit circuits, the inherent noise of the computer, the speed of the required gates, and the decoherence time of the qubits. Additional research could be done into the role of the verification to the security of the T22 protocol to define optimal number of circuits to check. This software simulation proves the benefit of the T22 QKD protocol in providing a secure key for encrypted communication.

VI. Acknowledgements

I would like to acknowledge my hackathon teammates Alexander Knapen and Julian Rice for their help in the initial implementation of the T22 protocol in an online secure chat website called Keytanglement. The Keytanglement hackathon project won second place in the QuTech

division of the MIT quantum hackathon, iQuHack, which took place in January 2022. I would also like to thank Dr. Katharina Gillen for supporting me and Alexander Knapen through a year-long independent study of various quantum information science topics and for her guidance through this project.

VII. Citations

[1] Brassard, Gilles, “A Brief History of Quantum Cryptography,” University of Montreal, October 17, 2005. [Online]. Available: https://arxiv.org/pdf/quant-ph/0604072.pdf. Accessed: June 8 2022.

[2] Song, D., Chen, D, "Quantum Key Distribution Based on Random Grouping Bell State Measurement," IEEE Communications Letters, vol. 24, no. 7, pp. 1496-1499, July 2020.

[3] Haitjema, Mart, “A Survey of the Prominent Quantum Key Distribution Protocols,” Washington University in St. Louis, December 2007. [Online]. Available: https://www.cse.wustl.edu/~jain/cse571-07/ftp/quantum/. Accessed: June 8 2022.

[4] Tiwari, Nayana, Knapen, Alexander, and Rice, Julian, “Bell State QKD,” Github, 2022. [Online] Available: https://github.com/nayanatiwari/bell-state-qkd. Accessed: June 1 2022.