# Quantum Information
# BB84 Quantum key distribution scheme

Barbara Noemi Szabo (s3263371), Ken Yeh (s2773430)
Group 6

April 26, 2024

## 1 Background

The goal of this project is to dive into quantum mechanics, a branch of physics that explores the behaviour of matter and energy at the most fundamental levels. By focusing on the foundational principles of quantum mechanics, this project aims not only to enhance our theoretical understanding but also to apply this knowledge in a practical context. Specifically, we will implement the simulation of quantum key distribution (QKD) protocol using the BB84 protocol, which represents a cutting-edge application of quantum mechanics in the field of secure communication. On top of that, we will also verify the result by implementing a middleman attack to prove the robustness of the protocol.

The implementation of the project can be found on Github. The link of the repository is in appendix A.

## 2 Theory

### 2.1 Definitions

**Qubit:** The basic unit in quantum information theory. In physics, when two distinguishable (and orthogonal) states are present, the qubit exhibits a phenomenon known as superposition. This implies that it is not in either of the two states individually, but rather in a combined state of the two. Only when we measure the qubit the superposition collapses and we find it in one of the states. To mathematically represent it first we define the two stats as $|0\rangle$ and $|1\rangle$.

$$|0\rangle \equiv \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad |1\rangle \equiv \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

A qubit is the linear combination of these states.

$$|q\rangle = \alpha \cdot |0\rangle + \beta \cdot |1\rangle, \quad \textit{where } |\alpha|^2 + |\beta|^2 = 1 \textit{ and } \alpha, \beta \in \mathbb{C} \tag{1}$$

$\alpha$ and $\beta$ are the amplitudes of the qubit. If we measure $|q\rangle$ the probability that we find it in state $|0\rangle$ is $|\alpha|^2$ and respectively, the probability that we find it in state $|1\rangle$ is $|\beta|^2$.

We can also write $|q\rangle$ as the following:

$$|q\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$$

To represent the bit 1 and 0 as a qubit we use the following mapping:

$$0 \to |0\rangle$$
$$1 \to |1\rangle$$

**Quantum gate:** A quantum gate performs an operation on a qubit which means that applying a quantum gate on a qubit is going to change its amplitudes ($\alpha$ and $\beta$). Mathematically it is represented as a unitary matrix, it guarantees that the property of the amplitudes ($|\alpha|^2 + |\beta|^2 = 1$) still holds after the operation. These gates are linear operations so applying the operation ($U$) to the qubit ($|q\rangle$) means the following matrix-vector multiplication:

$$\begin{aligned} U|q\rangle &= U(\alpha \cdot |0\rangle + \beta \cdot |1\rangle) \\ &= \alpha \cdot U|0\rangle + \beta \cdot U|1\rangle \end{aligned}$$

It means that to define a quantum gate we only have to show how it is applied for the basis states.

**Hadamard gate:** One of the most used quantum gates is the Hadamard gate ($H$).

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

Applying the Hadard gate for the basis states we get:

$$\begin{aligned} H|0\rangle &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \end{pmatrix} \\ &= \frac{1}{\sqrt{2}} \cdot \begin{pmatrix} 1 \\ 1 \end{pmatrix} \\ &= \frac{|0\rangle + |1\rangle}{\sqrt{2}} \end{aligned}$$

We usually refer to this state as $|+\rangle$.

$$\begin{aligned} H|1\rangle &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ &= \frac{1}{\sqrt{2}} \cdot \begin{pmatrix} 1 \\ -1 \end{pmatrix} \\ &= \frac{|0\rangle - |1\rangle}{\sqrt{2}} \end{aligned}$$

Similarly, we can refer to this state as $|-\rangle$.
In the following table, we summarise an important property of these states.

| State | $\alpha$ | $\beta$ | $\lvert\alpha\rvert^2$ | $\lvert\beta\rvert^2$ |
|---|---|---|---|---|
| $\lvert 0\rangle$ | 1 | 0 | 1 | 0 |
| $\lvert 1\rangle$ | 0 | 1 | 0 | 1 |
| $\lvert +\rangle$ | $1/\sqrt{2}$ | $1/\sqrt{2}$ | 0.5 | 0.5 |
| $\lvert -\rangle$ | $1/\sqrt{2}$ | $-1/\sqrt{2}$ | 0.5 | 0.5 |

Table 1: Summery of amplitude and probability values.

As a reminder $|\alpha|^2$ and $|\beta|^2$ are the probabilities to find the qubit in state $|0\rangle$ and $|1\rangle$ respectively after measurement.

One more useful property of the gate is that if we apply it twice for a basic state we get back itself.

$$\begin{aligned} H(H|0\rangle) &= H|+\rangle \\ &= H\left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right) \\ &= \frac{1}{\sqrt{2}}H|0\rangle + \frac{1}{\sqrt{2}}H|1\rangle \\ &= \frac{|0\rangle + |1\rangle}{2} + \frac{|0\rangle - |1\rangle}{2} \\ &= |0\rangle \end{aligned}$$

$$
\begin{aligned}
H(H|1\rangle) &= H|-\rangle \\
&= H\left(\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right) \\
&= \frac{1}{\sqrt{2}}H|0\rangle - \frac{1}{\sqrt{2}}H|1\rangle \\
&= \frac{|0\rangle + |1\rangle}{2} - \frac{|0\rangle - |1\rangle}{2} \\
&= |1\rangle
\end{aligned}
$$

**Basis:** In the scope of our project basis means a bit string where a 1 at index $i$ means applying the Hadamard gate for the $i^{th}$ element of the message and 0 means not applying.

**Shift:** To shift the message means to use a basis for a message and apply the Hadamard gate for the qubits in the message at all index $i$ where in the basis there is a 1.

## 2.2 QKD Basic model

Sharing a cryptographic key is the most important part of ensuring secure communication across digital channels. It serves to encrypt and decrypt messages. So it is essential for the parties to be able to agree on a key in a way that it remains hidden from the rest of the world. QKD solves this issue by transmitting quantum states between the two parties. This process involves encoding classical information onto quantum particles, typically photons. By leveraging the principles of quantum mechanics: manipulating the quantum states and exploiting the uncertainties of quantum measurements, QKD protocols guarantee the detection of any eavesdropping attempts, thereby ensuring the confidentiality of the exchanged information.

## 2.3 BB84

### 2.3.1 Overview

The BB84 protocol, introduced by Charles Bennett and Gilles Brassard in 1984, is a QKD protocol that allows two parties, traditionally named Alice and Bob, to generate a shared secret key, which can then be used for secure communication [1]. The protocol utilises the principles of quantum mechanics and the no-cloning theorem, to ensure the security of the key exchange process. In the BB84 protocol, Alice sends Bob a series of qubits encoded string in binary chosen at random. Bob then measures these qubits using a basis selected at random. After the transmission, Alice and Bob publicly compare their choice of bases. When their bases align, the corresponding bits form part of the secret key.

### 2.3.2 Steps

The protocol consists of two separate phases.

#### Message exchange phase

1. Alice generates the message as a sequence of bits ($A_{msg}$).
2. Alice generates a sequence of qubits representing the message ($A_{q\_msg}$).
3. Alice generates a random sequence of bases ($A_b$).
4. Alice shifts the message ($A_{q\_msg}$) using her bases ($A_b$) to get the shifted message ($A_s$).
5. Alice sends the shifted message ($A_s$) to Bob.
6. Bob receives the message ($B_r$).
7. Bob generates a random sequence of bases ($B_b$)
8. Bob use his bases ($B_b$) to shift the received message ($B_r$) to get his shifted message ($B_s$).
9. Bob measures the qubits from his shifted message ($B_s$) to get the final form of the message in bits ($B_{msg}$).

**Check phase**

1. Alice and Bob publicly share for which indices they used a basis shift (Hadamard gate). Because at these indices if there was no attack they must have the same values with probability 1.
2. $A_k$ and $B_k$ are the bit strings from $A_{msg}$ and $B_{msg}$ where they both used Hadamard gate or neither of them used.
3. They publicly share the values from the first half of $A_k$ and $B_k$.
4. If they have the same numbers that means that the probability that there was an attack is low and goes to zero if the length of the message goes to infinity. So the rest of $A_k$ and $B_k$ should be also equal so they can use the second half of the bit string (which they didn't share publicly) as a private key.

### 2.3.3 Example

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| $A_{msg}$ | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| $A_{q\_msg}$ | $\|0\rangle$ | $\|0\rangle$ | $\|1\rangle$ | $\|1\rangle$ | $\|0\rangle$ | $\|0\rangle$ | $\|1\rangle$ | $\|1\rangle$ |
| $A_b$ | **0** | **1** | **0** | **1** | 0 | 1 | 0 | 1 |
| $A_s$ | $\|0\rangle$ | $\|+\rangle$ | $\|1\rangle$ | $\|-\rangle$ | $\|0\rangle$ | $\|+\rangle$ | $\|1\rangle$ | $\|-\rangle$ |
| $B_r$ | $\|0\rangle$ | $\|+\rangle$ | $\|1\rangle$ | $\|-\rangle$ | $\|0\rangle$ | $\|+\rangle$ | $\|1\rangle$ | $\|-\rangle$ |
| $B_b$ | **0** | **1** | **0** | **1** | 1 | 0 | 1 | 0 |
| $B_s$ | $\|0\rangle$ | $\|0\rangle$ | $\|1\rangle$ | $\|1\rangle$ | $\|+\rangle$ | $\|+\rangle$ | $\|-\rangle$ | $\|-\rangle$ |
| $B_{msg}$ | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 1 |
| $A_k = B_k$ | 0 | 0 | 1 | 1 | | | | |

Table 2: Example of BB84 without man-in-the-middle attack.

## 2.4 No-cloning theorem

The "No cloning theorem" is a rule in quantum mechanics that says that you can't make an exact copy of an unknown quantum state. This is essential to ensure the security of BB84 protocol because it means that during a man-in-the-middle attack the attacker - usually refferd to as Eve - can't measure a quantum state (a qubit) received from Alice and send the same state to Bob, hence Eve can't go unnoticed. The prrof of the theorem can be found in appendix B [2].

## 2.5 Adversary Strategy

In the implemented adversary strategy Eve receives the qubits that Alice wants to send to Bob and measures them. Eve can't copy (clone) the unknown states and send them directly to Bob because of the No-cloning theorem so she applies a random bases on the measurements and sends this sequence to Bob.

## 2.6 The failure of the protocol

In defining the failure criteria for BB84 in the context of man-in-the-middle attacks, we establish three distinct scenarios.

Firstly, if the protocol incorrectly detects an attack when none has occurred, leading the participants to believe their communication is compromised, thereby resulting in an unsuccessful key distribution.

Secondly, if the protocol fails to detect an actual attack, leaving the participants unaware of the intrusion. In this latter case, although Alice and Bob may assume their communication is secure, but they don't necessarily share the same key and Eve, the malicious intermediary, gains access to their keys. Armed with this information, Eve can decrypt their messages, eavesdrop on their communication, and even manipulate the exchanged information, posing a significant threat to the confidentiality and integrity of their correspondence.

Finally, if the protocol correctly detects that there was no eavesdropper but the length of the shared key is zero.

# 3 Implementation

For the implementation of the simulation, we employ two distinct methodologies. The first approach relies on an intuitive strategy, where we closely adhere to the procedural guidelines outlined in the referenced academic

paper. This method allows us to replicate the process straightforwardly. The second approach takes a more rigorous path, utilising formal mathematical techniques to elucidate and verify the underlying mechanisms of the simulation. They will be further explained in this section. One can also check the detailed implementation by accessing the repository created for this project as Appendix A indicates.

### 3.1 Intuitive approach

For this intuitive approach, we implemented the protocol by applying the steps we defined in section 2.3.1 in combination with the paper as a reference [3]. Some variables are defined:

- $n$: Number of qubits in the final key.
- $\delta$: A small value is added to the $n$ for the protocol to work correctly and also for error control.
- $length$: Length of the message. According to the paper, it is defined as $n \cdot (4 + \delta)$.

To ensure the protocol works correctly in this approach, we also implemented some additional checks:

1. The output length of the sifted key should not be more than $length$.
2. Sifted key should only contain 0s and 1s.

### 3.2 Formal approach

In this approach, we also followed the steps of the protocol but implemented it by using the mathematical representations of qubits and quantum gates.

For this reason, we have a class called Qubit that can model a qubit state with its amplitudes and it makes sure that the criteria defined in Equation 1 hold. It can calculate the exact probabilities from the amplitudes or estimate them by simulating measurements multiple times in the same state. Applying the Hadamard gate was also implemented by matrix-vector multiplication:

```
def hadamard(self):
        # Definition of Hadamard gate
        H = 1/sqrt(2) * np.array([[1, 1], [1, -1]])

        # Apply Hadamard gate on the state
        new_amplitdes = (H @ self.vec)
        self.update_amplitude(*new_amplitdes)
```

Listing 1: Hadamard gate

Then we modeled the protocol by following the steps defined in Section 2.3.1 with class named BB84. For running the protocol it separates the message exchange and the checking phases then checks whether the protocol failed or not using the conditions form Section 2.5.

## 4 Results

We utilise the same method for both approaches to gather and verify results. That is, calculate the probability of failure on different message lengths. For failure, we are referring to what we defined in 2.5. For both approaches, we ran the protocol 10,000 times per length (from 1 to 24) for the case with a middleman attack and without then calculating the relative frequency of failure to approximate the probabilities.

For the intuitive approach, we have the following:

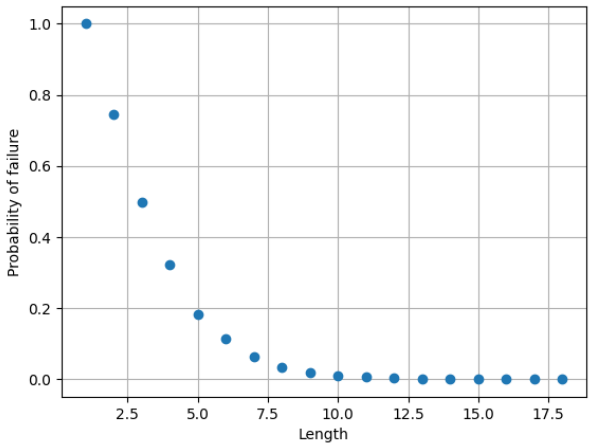


Figure 1: Probability of with no attack for different message lengths

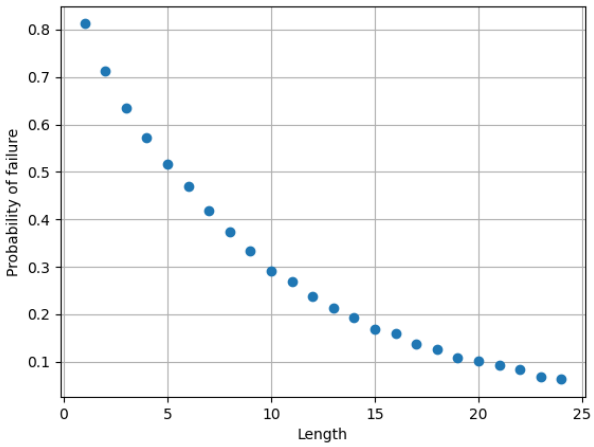


Figure 2: Probability of with attack for different message lengths

For the formal approach, we have the following:

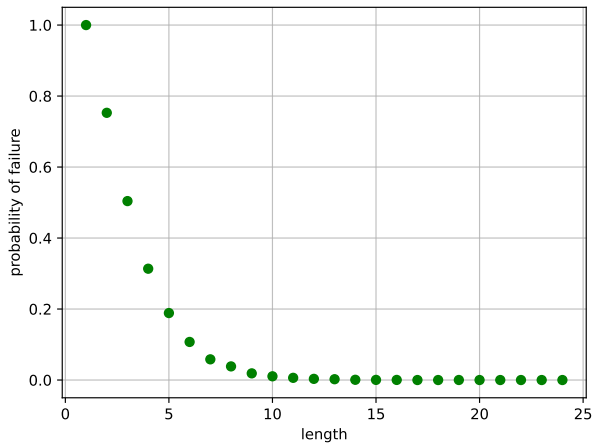


Figure 3: Probability of with no attack for different message lengths

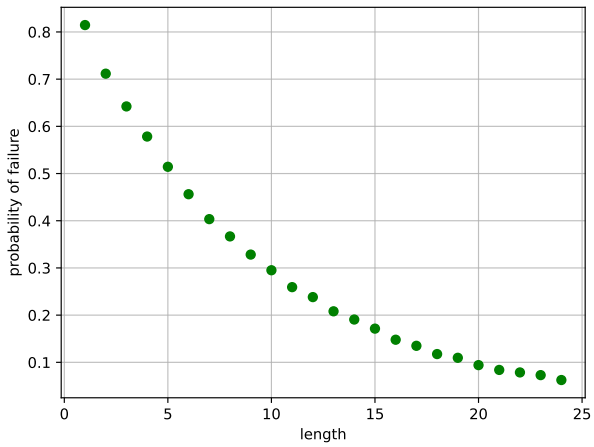


Figure 4: Probability of with attack for different message lengths

We were able to get the same results with both approach.

From the results, one can observe that when the length of the message increases, the protocol is less likely to fail, moreover it looks like it is going to zero if the length of the message goes to infinity. This shows the robustness of the BB84 protocol.

# 5 Conclusion

In conclusion, we have implemented the BB84 protocol using two different approaches and have shown the reliability of the protocol based on the observation of the probability of failure on different message lengths. Both of the approaches obtained the same outcome and showed the same pattern of behavior of the BB84 protocol.

# References

[1] C. H. Bennett and G. Brassard, "Quantum cryptography: Public key distribution and coin tossing," *Theoretical Computer Science*, vol. 560, p. 7–11, Dec. 2014. [Online]. Available: http://dx.doi.org/10.1016/j.tcs.2014.05.025

[2] P. E. Black, D. R. Kuhn, and C. J. Williams, “Quantum computing and communication,” ser. Advances in Computers, M. V. Zelkowitz, Ed. Elsevier, 2002, vol. 56, pp. 189–244. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0065245802800079

[3] P. W. Shor and J. Preskill, “Simple proof of security of the bb84 quantum key distribution protocol,” *Physical Review Letters*, vol. 85, no. 2, p. 441–444, Jul. 2000. [Online]. Available: http://dx.doi.org/10.1103/PhysRevLett.85.441

# Appendix

## A Source Code

All source code used in this project can be found in the repository with the following link: GitHub

## B Proof of No-Cloning Theorem

Let's assume that with operator $U$ we can clone an unknown state and $\otimes$ denotes the tensor product. Consider two orthogonal states $|\psi\rangle$ and $|\phi\rangle$. From the definition of cloning

$$U(|\psi\rangle \otimes |0\rangle) = |\psi\rangle \otimes |\psi\rangle)$$
$$U(|\phi\rangle \otimes |0\rangle) = |\phi\rangle \otimes |\phi\rangle)$$

$U$ is a quantum gate, so it is a linear operator, thus

$$\begin{aligned} U[(\alpha|\psi\rangle + \beta|\phi\rangle) \otimes |0\rangle] &= \alpha U(|\psi\rangle \otimes |0\rangle) + \beta U(|\phi\rangle \otimes |0\rangle) \\ &= \alpha|\psi\rangle \otimes |\psi\rangle + \beta|\phi\rangle \otimes |\phi\rangle, \quad \textit{Def. of cloning} \end{aligned}$$

But if we directly apply the definition of cloning and the distributive property of tensor product:

$$\begin{aligned} &U[(\alpha|\psi\rangle + \beta|\phi\rangle) \otimes |0\rangle] \\ &\quad = (\alpha|\psi\rangle + \beta|\phi\rangle) \otimes (\alpha|\psi\rangle + \beta|\phi\rangle) \\ &\quad = \alpha^2|\psi\rangle \otimes |\psi\rangle + \alpha\beta|\psi\rangle \otimes |\phi\rangle + \alpha\beta|\phi\rangle \otimes |\psi\rangle + \beta^2|\phi\rangle \otimes |\phi\rangle \end{aligned}$$

The two results are only equal if:

$$\alpha^2 = \alpha \qquad \alpha\beta = 0 \qquad \beta^2 = \beta$$

So either $\alpha$ or $\beta$ has to be 0.