Individual report 0031

Zican Wang

I. Introduction

In this project, the aim is to optimize the key rate for the two-state quantum key distribution protocol[1] according to the disturbance in the quantum communication channel. The result of this project shows a decreasing trend for the key rate when the disturbance gets higher and additional relationships between the elements affecting the key rate are looked at. The best angle for maximising the key rate is also computed for each value of disturbance.

II. Personal contribution

A. Representations

To comprehend QKD protocols, I needed to have a firm grasp on the fundamentals of quantum computing and its representations in order to benefit from related publications. In classical computation, information is represented by strings of 0's and 1's. In quantum computation and information, the 0's and 1's (computational basis) can be one of numerous bases. Vectors can be used to represent between 0 and 1, as well as all other strings derived from the computational basis. In quantum computation, the unit of computation is referred to as a 'qubit,' in contrast to the 'bit' in classical computation. The vector denotes the state of a qubit, while the binary denotes the state of a bit. In contrast to the bits in the computational basis, all unit vectors are valid states of a qubit, which means that vectors representing qubits can have more than one non-zero element.

$$0 \to \begin{bmatrix}1\\0\end{bmatrix}, 1 \to \begin{bmatrix}0\\1\end{bmatrix}, x \to \begin{bmatrix}0\\0\\\vdots\\1\\\vdots\end{bmatrix} \begin{matrix}\text{'...00'}\\\text{'...01'}\\\vdots\\\text{'x in binary'}\\\vdots\end{matrix}$$

Thus, the computational basis can be thought of as a vector span with a horizontal vector of 0 and a vertical vector of 1. Other bases form spans that are to some angle of the computational basis (Figure 1). All unit vectors (qubits) trace out a circle indicated in orange.

The vector notation of bits can be abbreviated by Dirac notation, this is simply adding '|' and '>' around the bit value. Adding '<' and '|' means the conjugate transpose of the column vector the bit value represents.

$$|\psi\rangle = \begin{bmatrix}\alpha\\\beta\end{bmatrix}, \langle\psi| = \begin{bmatrix}\alpha^*, \beta^*\end{bmatrix}$$

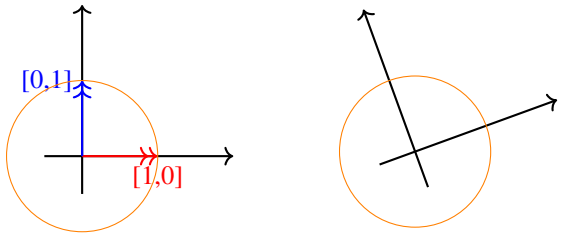

Fig. 1: Different bases

This way, superpositions can be easily expressed using additions and subtractions of vectors. The concatenations of bits are represented by '$\otimes$' named 'tensor products' and sometimes it can be omitted, for example, $|10\rangle = |1\rangle \otimes |0\rangle = |1\rangle\,|0\rangle$.

B. Quantum measurements

Quantum measurements, unlike measurements in the classical physics world, can cause a quantum particle or qubit to change its state, but classical measurements do not have a detectable effect on the object being measured.[2].

In brief, the original state of some qubits can be superpositions – each string has a probability of being the measurement outcome, shown by the square of the corresponding element in the state vector. Thus, the probability of getting a string x (computational basis) when measuring ψ is calculated by $\langle x| \times |\psi\rangle$ or $\langle x|\psi\rangle$ in short. The state of those qubits after measuring collapses to the outcome we get with fix probability 1. This is often referred to as the measurement postulate or Born's rule.

For example, the state $|\psi\rangle = \begin{pmatrix}1/\sqrt{2}\\1/\sqrt{2}\end{pmatrix}$, when being measured in the computational basis, has probability $(\frac{1}{\sqrt{2}})^2 = 50\%$ of getting the outcome 0 or 1. After the measurement, the state of ψ becomes 1 if the outcome is 1, and becomes 0 otherwise (Figure 2). Also note that $|\psi\rangle$ can be a linear combination of the two basis vectors, which infers that a state vector can be represented by a linear combination of the vectors in another non-orthogonal basis. Thus, linear combinations can be comprehended as superpositions and in this example, $|\psi\rangle$ is a superposition of 0(1,0) and 1(0,1). $|\psi\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$.

From this, we can see that measuring a state using a basis non-orthogonal to the state's basis will give more than one possible outcome and result in a change of the state.

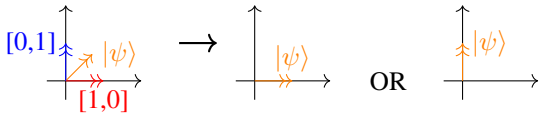

Fig. 2: measuring $|\psi\rangle$

C. Understanding QKD protocols

Quantum measurements are critical in ensuring the security of QKD methods. Due to the fact that measuring a qubit alters its state, QKD techniques use qubits to represent the bits in the keys. The outcome is fixed if the qubit is measured on the same basis as it was sent. However, if the measuring and sending bases are at an angle, the result is probabilistic. Both entrusted parties can check for changes after sending the keys, and the difference would signal that a third party measured or interacted with the qubit. We'll refer to the sender as Alice, the receiver as Bob, and the opponent as Eve.

As a prequel to this project, one other QKD protocol is looked at, which is the first proposal of key distribution using theories of quantum mechanics [3]. By analysing its behaviour, I discovered that the QKD protocol makes use of two basis that have some angle to each other but are not orthogonal. This way, only the sender knows which of the two basis is chosen, and if the receiver need to "guess" the basis that is chosen, there is a significant probability that the receiver measures the qubit on a different basis and therefore changes its state to be one of the basis vectors of the different basis. Consequently, Bob and Alice can check with each other, for each qubit sent, the matched basis that they choose after the qubits are sent, and Bob knows that the qubits measured on this basis are correct and he will discard all the bits measured with the wrong basis. Alice and Bob will then choose a sample part of the remaining key to check if they are the same. Because the attacker also need to guess the basis used to send the qubit and a wrong guess would alter the state. A difference would imply interference between two parties.

There must be an error correction phase[4] that corrects all of the inconsistencies between the keys that each party holds in their possession. To verify for total discrepancies, the entrusted parties would use a function that generates a unique sequence based on their respective keys and would then check with each other without leaking information. Then a $Universal_2$ function is sent from Alice to Bob[5] and Bob will use it to correct the errors.

After the error correction part of the protocol, the key requires another post-processing stage to strengthen its security. This stage tries to minimise Eve's knowledge of the final key. A $Universal_2$ function is also used here. The function is published and used by both parties to remove bits from the key so that the final key has no correlation with Eve's information.

D. Analysing the two-state protocol

The project works on the two-state protocol, which has a lower disturbance tolerance than the BB84 protocol[6]. In general, the post-processing phases only require classical computation, and they are compatible with both protocols. The difference lies in the way the initial key is distributed. For the two-state protocol, Alice sends only two types of qubits, one from each non-orthogonal basis, instead of the four types in the BB84 case. Since there is only one qubit generated from each basis, each basis can be treated as representing a bit value of 0 or 1, and Bob only needs to know which basis is used when generating the qubits. The two basis shown in figure 3 are Bob's measuring basis, with each basis constructed by two basis vectors. Since the state vectors of Alice's qubits are one of the basis vectors of each basis individually, Bob will always receive the unaltered qubit that Alice sent if he measures on the basis that has the qubit's state vector as its basis vector. This means that if Bob gets the other basis vector from one basis, he can deduce that Alice sent the qubit that is not a basis vector of his basis.

For the distribution phase, Alice generates a random number $x_i \in \{0, 1\}$ with probability $P(x_i) = 1/2$, then prepares the state:

$$|\varphi_{x_i}\rangle = \cos\theta\,|0\rangle + (-1)^{x_i}\sin\theta\,|1\rangle \ , \tag{1}$$

and sends it to Bob. Bob generates a random number $y_i \in \{0, 1\}$ with probability $P(y_i) = 1/2$ and performs a measurement in the basis $\{|\varphi_{y_i}\rangle, |\bar{\varphi}_{y_i}\rangle\}$, where we define

$$|\bar{\varphi}_{y_i}\rangle = \sin\theta\,|0\rangle - (-1)^{y_i}\cos\theta\,|1\rangle \ . \tag{2}$$

- If the outcome is $|\bar{\varphi}_{y_i}\rangle$ then Bob's guess for x_i is $x_i' = 1 - y_i$.
- If the outcome is $|\varphi_{y_i}\rangle$ then Bob tells Alice to discard this round.

The generated data is (x_j, x_j') for $j = 1, \ldots, N'$.

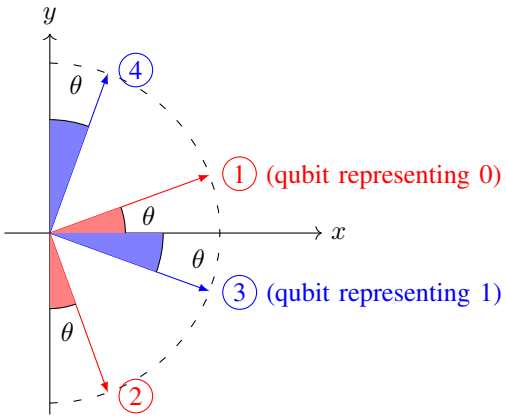

Fig. 3: state vectors and the bases that Bob would use, The red basis can only get 3 and the blue basis can only get 1. For example, for the red basis, if Alice sent 1 and Bob uses red to measure, he will always get 1, the only occasions that he can get 2 is when Alice uses 3

For the two-state protocol in question, the key factors for each stage are labelled with abbreviations for calculation and explanation. Initially, the transferred key is N in length and the disturbance in the channel is D. The value D is calculated by Alice and it is the rate of correct bits when they sample a portion of the received key and check with each other. Bob's remaining key has a length of N_r after he discarded all the bits about which he was unsure.The final key length after all the post-processing parts is N_k. The following equation describes the relationship between the final key length and the initial key length:

$$N_{\mathrm{k}} = \lfloor N_{\mathrm{r}}(H(X|E) - h(D)) \rfloor \tag{3}$$

where $h(p)$ is the binary entropy of a probability p and it is defined as:

$$h(p) = -p\log_2 p - (1-p)\log_2(1-p) \tag{4}$$

$H(X|E)$ denotes the attacker's ignorance of the key, which is determined by factors that are not controllable by Alice or Bob. Thus, the optimization should consider this in the worst case scenario where $H(X|E)$ is minimized. Given that the only factor that can be changed by Alice is the angle θ, which has an effect on the key rate, we may be able to express N_k as a function of θ.

E. Related works

Other projects related to this one are looked at for comparison and guidance. According to one study, the key rate falls as the disturbance increases, with the best tolerance attained at roughly 10%. As a result, the initial test for disturbance would be confined to values between 0 and 0.2[6]. Other publications discuss possible optimization strategies for an extended version of the two-state protocol [7]. The optimization is performed using the Mathematica software for these projects.

F. Calculating the key rate and deriving the entropy

The objective is to derive the final key length N_k to some function of θ and determine the angle that results in the longest key. According to figure 3 and table I, the angle between x_i and $y_i = 1 - x_i$ is $\frac{\pi}{2} - 2\theta$, thus the probability of getting $|\bar{\varphi}_{y_i}\rangle$ for each pair of prepared state and measuring basis is $\cos^2(\frac{\pi}{2} - 2\theta)$. Bob has a probability of a half of getting the right basis that will produce a valid bit, so the overall rate is $\frac{1}{2}\cos^2(\frac{\pi}{2} - 2\theta)$.

y_i	$\lvert\varphi_{y_i}\rangle$	$\lvert\bar{\varphi}_{y_i}\rangle$
0	$\cos\theta\lvert 0\rangle + \sin\theta\lvert 1\rangle$ (1)	$\sin\theta\lvert 0\rangle - \cos\theta\lvert 1\rangle$ (2)
1	$\cos\theta\lvert 0\rangle - \sin\theta\lvert 1\rangle$ (3)	$\sin\theta\lvert 0\rangle + \cos\theta\lvert 1\rangle$ (4)

TABLE I: Bob's measuring basis according to y_i

Hence if there are no disturbances and the number of bits in the beginning is N, then afterwards Bob will have $N' = \frac{1}{2}\cos^2(\frac{\pi}{2} - 2\theta) \times N$ bits left.

1) Partial measurement: However, with the presence of an attack or channel noise, the rate of the keys would change. All attacks and interference can be treated as the most general interaction between another qubit and the sender qubit. We write this general interaction as:

$$\mathcal{U}(|0\rangle_E \otimes |\varphi_x\rangle_B) = |\Phi_x\rangle_{EB} = \sum_{r,s=0}^{1} \Gamma_x^{r,s} |r\rangle_E |s\rangle_B \ . \tag{5}$$

Bob then does a partial measurement without the interacting qubit. The probability now would depend on the interacting qubit, and would become conditional.

2) Probability change: The length of Bob's key N' would be the probability that he accepts the qubit:

$$\begin{aligned} \mathrm{prob}\{\bar{\varphi}\} =& \frac{1}{4} \times \| \langle\bar{\varphi}_0|\Phi_0\rangle \|^2 + \frac{1}{4} \times \| \langle\bar{\varphi}_0|\Phi_1\rangle \|^2 + \\ & \frac{1}{4} \times \| \langle\bar{\varphi}_1|\Phi_0\rangle \|^2 + \frac{1}{4} \times \| \langle\bar{\varphi}_1|\Phi_1\rangle \|^2 \\ =& \frac{1}{4} \sum_{x,y} \| \langle\bar{\varphi}_y|\Phi_x\rangle \|^2 \end{aligned} \tag{6}$$

Later on, we will utilise Bob's probability of getting the correct qubit, which actually means that it is conditioned on him accepting the qubit in the first place, as shown in table II. Initially, I attempted to use the probability of getting the right probability with no condition, but this resulted in getting an invalid outcome when calculating the density matrices for Eve, where the probabilities do not add up to one.

$H(X|E)$ is given by:

$$H(X|E) = H(E|X) + H(X) - H(E) \tag{7}$$

Each term of the left hand side is derived in to functions of θ and Γ. Both $H(E|X)$ and $H(E)$ are calculated using density matrices $\sigma_{E|X}$ and σ_E.

Let us define Eve's (normalised) conditional states $|\mu_{x',x}\rangle_E$ as:

$$\sqrt{q(x'|x)}\,|\mu_{x',x}\rangle_E = {}_B\langle\bar{\varphi}_{1-x'}|\Phi_x\rangle_{EB} \tag{8}$$

for $x', x \in \{0,1\}$. Since $|\mu_{x',x}\rangle_E$ is normalized, $\sqrt{q(x'|x)}^2 = q(x'|x)$ is the probability that Bob would get x'. This is calculated by matrix multiplication and then used to construct the density matrices. There are four different $q(x'|x)$ values for each value of x' and x's. The two density matrices $\sigma_{E|X}$ are each constructed from $q(x'|x)$ with the same x. The terms are mostly used repeatedly and $q(x'|x)$s are made up of these components:

- $\sin\theta\Gamma_x^{0,0}$
- $\cos\theta\Gamma_x^{0,1}$
- $\sin\theta\Gamma_x^{0,0}$
- $\cos\theta\Gamma_x^{1,1}$

These are used in combination for functions for $H(X|E)$.

Alice's basis	Bob's receiving basis			
	y = 0		y = 1	
	Bob's receiving state		Bob's receiving state	
	$\|\varphi_{y_0}\rangle$	$\|\bar{\varphi}_{y_0}\rangle$	$\|\varphi_{y_1}\rangle$	$\|\bar{\varphi}_{y_1}\rangle$
x = 0	reject	accepted but wrong	reject	accept and correct
x = 1	reject	accepted but correct	reject	accept but wrong

TABLE II: Bob's measured bits corresponding to Alice's and Bob's basis. Alice has an uniform distribution of choosing x, and so does Bob when choosing basis number y. Bob will not accept any bits when he gets the state $|\varphi_y\rangle$, and he will accept all bits with state $|\bar{\varphi}_y\rangle$. Notice that when Bob receives $|\bar{\varphi}_y\rangle$, he calculates the bit $x' = 1 - y$. Thus so when $y = x$, Bob will get the wrong state as what Alice has prepared.

G. Calculating the optimization

1) constraints: There are constraints on the $q(x'|x)$ values so that optimization functions can converge. There are three constraints that can be derived. The first two are because of the unitary interaction:

$$\langle \Phi_x | \Phi_x \rangle = 1 \ , \tag{9}$$

$$\langle \Phi_0 | \Phi_1 \rangle = \cos 2\theta \ . \tag{10}$$

The third one is a consequence of the disturbance D, which also depends on θ and Γ.

$$D = \frac{\text{prob}\{\textbf{accepted and correct}\}}{\text{prob}\{\textbf{accepted}\}} \tag{11}$$

$$= \frac{q(0|1) + q(1|0)}{q(0|0) + q(0|1) + q(1|0) + q(1|1)} \tag{12}$$

$$= \frac{q(0|1) + q(1|0)}{\sum_{x',x} q(x'|x)} \tag{13}$$

These three constraints all need to be satisfied when optimizing the key rate.

2) Adding N' to the formula: As mentioned previously, the final key rate can be evaluated as a function of θ and Γ, with Alice as the sender choosing the value of θ and Eve determining the value of Γ. We want to minimize $H(X|E)$ so that Eve learns as mush knowledge of the key as possible. Another factor that alters the final key rate is the length of the key after Bob discards the rejected qubits, N'. The N_r here is omitted as the difference between it and N' is negligible when N is sufficiently large. By appending N' to the function representation of N_k, the following is implied:

$$N_{\text{k}} = \lfloor N'(H(X|E) - h(D)) \rfloor \tag{14}$$

Then, plots with axis D, θ and final key rate should be generated, with the optimum angle being the maximum point of the key rate for a given D.

H. Coding

The optimization calculation uses the Wolfram Mathematica software. I tried some abbreviations for some terms that are often used in combination figure4. In this code:
$a_x = \sin\theta\Gamma_x^{0,0}$, $b_x = \cos\theta\Gamma_x^{0,1}$,
$c_x = \sin\theta\Gamma_x^{1,0}$, $d_x = \cos\theta\Gamma_x^{1,1}$.
Due to the fact that the Γs are always used with the same trigonometric functions, their forms are shortened using capital letters.

In addition, as referred to in the calculation part in both reports, the density matrix σ is used, and the majority of the time it is used for calculating the entropy, necessitating the abbreviation of the diagonal values. The specifics are depicted in figure 4.

```
A0 = Sin[θ] x a0;
B0 = Cos[θ] x b0;
C0 = Sin[θ] x c0;
D0 = Cos[θ] x d0;
A1 = Sin[θ] x a1;
B1 = Cos[θ] x b1;
C1 = Sin[θ] x c1;
D1 = Cos[θ] x d1;

sige000 = (A0^2 + B0^2) / (A0^2 + B0^2 + C0^2 + D0^2);
sige011 = (C0^2 + D0^2) / (A0^2 + B0^2 + C0^2 + D0^2);
sige100 = (A1^2 + B1^2) / (A1^2 + B1^2 + C1^2 + D1^2);
sige111 = (C1^2 + D1^2) / (A1^2 + B1^2 + C1^2 + D1^2);
```

Fig. 4: abbreviating terms that is used more than once

The entropy values that I used to construct $H(X|E)$ is represented using the abbreviated terms, shown in figure 5. $H(X)$ is equal to one in this case because the probability is uniformly distributed for qubits representing 1's and 0's. The disturbance, D, and the key length from the intermediate stages are also defined there. $N' = N_r$ is used since the difference between the two values can be negligible, as explained in section II-G. Afterwards, $H(X|E)$ is minimised for each value of θ and D, and the result is added to a list. The interval chosen for the increment of θ is 0.01 and the range of θ is 0 to $\frac{\pi}{4}$. This is because when $\theta = \frac{\pi}{4}$, then $2\theta = \frac{\pi}{2}$, which means that the basis will overlap, as shown in figure 3 and further increment would wrap around zero. The disturbance is set to values between 0 and 0.2 with a 0.005 increment.

Finally, a three-dimensional graph is generated to illustrate the relationship between D, θ and the final key rate. Negative values indicate that no effective key has been produced, and hence their key rate values have been changed to zero. Figure 7 shows how the graphs are generated and also includes code for two additional plots that depict relative relationships between each pair of the three axes.

```
HEX = (-sige000 × Log2[sige000] - sige011 × Log2[sige011])/2 + (-sige100 × Log2[sige100] - sige111 × Log2[sige111])/2;
HE = -((sige000 + sige100)/2) × Log2[(sige000 + sige100)/2] - ((sige011 + sige111)/2) × Log2[(sige011 + sige111)/2];

HXE = HEX + 1 - HE;

Disturbance = ((A0 - B0)^2 + (C0 - D0)^2 + (A1 + B1)^2 + (C1 + D1)^2) /
    ((A0 + B0)^2 + (C0 + D0)^2 + (A0 - B0)^2 + (C0 - D0)^2 + (A1 + B1)^2 + (C1 + D1)^2 + (A1 - B1)^2 + (C1 - D1)^2);
Nprime = 1/4 ((A0 + B0)^2 + (C0 + D0)^2 + (A0 - B0)^2 + (C0 - D0)^2 + (A1 + B1)^2 + (C1 + D1)^2 + (A1 - B1)^2 + (C1 - D1)^2);
Nr = Nprime;
M = Nr × (-Dis × Log2[Dis] - (1 - Dis) × Log2[1 - Dis]);
```

Fig. 5: representing the entropy and intermediate key lengths

```
l = {};
Do[
 l = Append[l, {θ, Dis, NMinimize[{(Nr × HXE - M),
                                   a0^2 + b0^2 + c0^2 + d0^2 == 1,
                                   a1^2 + b1^2 + c1^2 + d1^2 == 1,
                                   a0 × a1 + b0 × b1 + c0 × c1 + d0 × d1 == Cos[2 × θ],
                                   Dis == Disturbance},
                                  {a0, b0, c0, d0, a1, b1, c1, d1},
                                  WorkingPrecision → 15][[1]]}
           ],
 {θ, 0.01, π/4, 0.01}, {Dis, 0.01, 0.2, 0.005}]
```

Fig. 6: minimizing $H(X|E)$

```
ListPlot3D[l, AxesLabel → {"θ", "D", "Key rate"}, ImageSize → Large]

Nonzerol = {};
Do[Nonzerol = Append[Nonzerol, If[l[[i]][[3]] < 0, ReplacePart[l[[i]], 3 → 0], l[[i]]]], {i, 1, Length[l]}]


ListPlot3D[Nonzerol, AxesLabel → {"θ", "D", "Key rate"}, PlotRange → All, ImageSize → Full]

newlist = Gather[Nonzerol, Indexed[#1, 2] == Indexed[#2, 2] &];
plotlist = {};
Do[
  Dlist = Transpose[newlist[[i]]][[2]];
  Rlist = Transpose[newlist[[i]]][[3]];
  x = TakeLargest[Rlist, 1];
  plotlist = Append[plotlist, {Dlist[[1]], x[[1]]}], {i, 1, Length[newlist]}];
plotlist

ListPlot[plotlist, AxesLabel → {"D", "Key rate"}]


plotlist2 = {};
Do[
  newlist2 = Transpose[Transpose[newlist[[i]]]];
  newlist3 = MaximalBy[newlist2, Last];
  plotlist2 = Append[plotlist2, {newlist3[[1]][[3]], newlist3[[1]][[1]]}], {i, 1, Length[newlist]}];
plotlist2
ListPlot[plotlist2, AxesLabel → {"key rate", "θ"}]
```

Fig. 7: post-processing that clears the negative key rates and plots 2D graphs for each pair of variables

III. Project Assessment

A. Initial expectations

The final key rate is determined by the channel noise and the basis angle. We can deduce that higher channel noise would lead to a reduced percentage of successful qubit transfers, thus lowering the key rate. According to the related work, the reduction would be near linear, so a linear correlation is expected for the result.

In the beginning, the programme was set to maximise $H(X|E)$ because it is part of the function representing the final key rate. I reasoned that in order to maximise the key rate, we should maximise the values used to construct it. However, the plotted graph for the maximised $H(X|E)$ produced a straight line, which is inconsistent with the expectation. This is because $H(X|E)$, like D, is a constant as Alice send the qubits. Additionally different $H(X|E)$ values may give different key rate results. Here we assume the worst case where $H(X|E)$ is lowest for each value of D and θ and optimise the key rate from there.

B. Final results

The relationship between the key rate and the disturbance is illustrated in the first plot of figure 8. This is not as expected as the key rate decreases drastically over the initial increase of D and the steep curve becomes shallow as the disturbance reaches 0.05. What is also different is that there are still effective key rates when the disturbance reaches 10%, which is higher than the related projects in the research.

In addition, figure 8 illustrates another relationship that has not been depicted in previous publications. The second graph shows which value Alice should choose to obtain the optimal key rate with a known disturbance.

C. Scope for improvements

1) Including N_r values: One additional detail can be included when calculating the final key rate. Because the N_r value was deemed negligible, it was not included in the final calculation. An improvement may take account of the decreased N_r instead of using N'. The solution may become more realistic and close to the one demonstrated in the related work using finite qubits[7].
2) Trying different values for $H(X|E)$: In this project, $H(X|E)$ was set to the lowest value possible for different values of disturbances. We can also investigate the relationship between the entropy values and the key rate and generate multiple 3D plots for comparison.
3) Minimizing only the $H(X|E)$: When the final key rate is calculated by minimising the $H(X|E)$ values, the N_r value is also minimised in the equation. On the other hand, $N' \approx N_r$ is affected solely by the angle, which is not determined by the attacker. Thus, minimising only the entropy value is sufficient for the calculation, although the calculation would not change significantly because the angle and the disturbance are fixed values for each round of calculation.

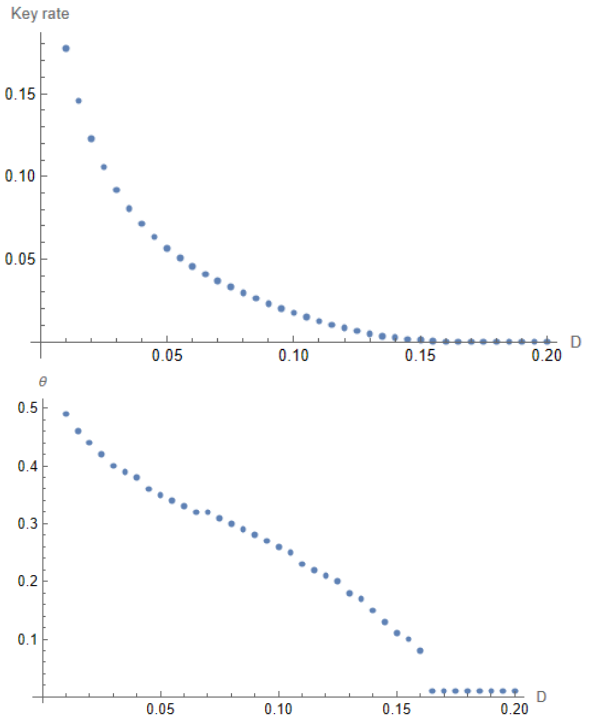

Fig. 8: correlations between D, the key rate and θ

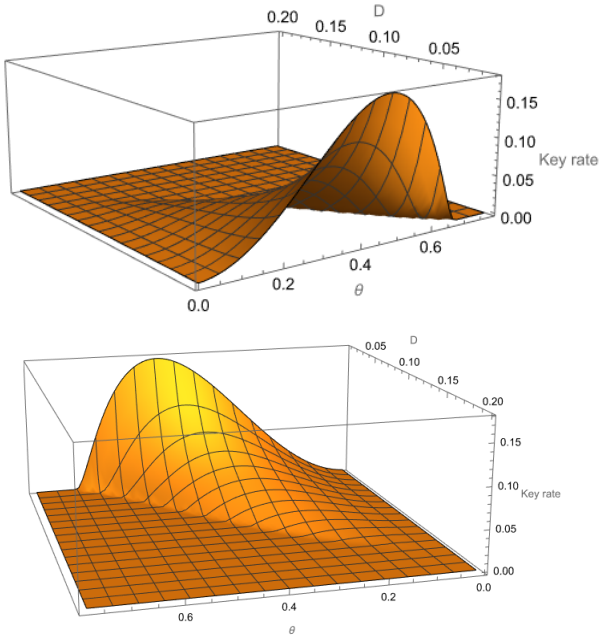

Fig. 9: The final plot with only non-negative values

IV. Team Assessment

A. Critical assessment of team overall

Since this course is a continuation of the individual research project COMP0030, this group consists of only one member due to the choice of topic. As a result, an overall team assessment is unnecessary.

B. Critical assessment of each member

1) Zican Wang: As the only member in the group, all contributions were done by myself, from the initial research to the final coding and optimisation to drafting the entire group report. The following are some critical stages for this project: QKD research, function construction, the calculation of $H(X|E)$ and coding.

When I started this project, I had a hard time understanding the notations and how the equations relate to each other. As I progressed through the research, I gained a deeper understanding of the fundamental meaning of what vectors, matrices, and entropy represent in the quantum world. I was focused on performing each calculation accurately and being honest about what I knew and what I did not know. I was unfamiliar with these types of representations, which slowed down the progress, but I remained patient and did not get eager for progress when doing the calculations. I was also really versatile between all the tasks in this project.

However, I can also be overly detail-oriented, to the point that I almost lose sight of the overall objective of optimising the angle. I was not flexible when performing the calculations in the beginning, but when I got used to doing these calculations, I would occasionally become insecure about my results.

REFERENCES

[1] Charles H Bennett. Quantum cryptography using any two nonorthogonal states. *Physical review letters*, 68(21):3121, 1992.

[2] Daniel E Platt. A modern analysis of the stern–gerlach experiment. *American Journal of Physics*, 60(4):306–308, 1992.

[3] Charles H Bennett and Gilles Brassard. Quantum cryptography: Public key distribution and coin tossing. *arXiv preprint arXiv:2003.06557*, 2020.

[4] Tor Helleseth. *Advances in Cryptology–EUROCRYPT'93: Workshop on the Theory and Application of Cryptographic Techniques Lofthus, Norway, May 23–27, 1993 Proceedings*, volume 765. Springer, 2003.

[5] Charles H Bennett, Gilles Brassard, and Jean-Marc Robert. Privacy amplification by public discussion. *SIAM journal on Computing*, 17(2):210–229, 1988.

[6] Hiroaki Sasaki, Ryutaroh Matsumoto, and Tomohiko Uyematsu. Key rate of the b92 quantum key distribution protocol with finite qubits. In *2015 IEEE International Symposium on Information Theory (ISIT)*, pages 696–699. IEEE, 2015.

[7] Omar Amer and Walter O Krawec. Finite key analysis of the extended b92 protocol. In *2020 IEEE International Symposium on Information Theory (ISIT)*, pages 1944–1948. IEEE, 2020.