Innovative Eavesdropper Attacks on Quantum Cryptographic Systems

Elizabeth Newton
Department of Physics and Astronomy
University of Leeds

A thesis presented for the degree of
Doctor of Philosophy
31st October 2018

The candidate confirms that the work submitted is their own, except where work which has formed part of jointly-authored publications has been included. The contribution of the candidate and the other authors to this work has been explicitly indicated below. The candidate confirms that appropriate credit has been given within the thesis where reference has been made to the work of others.

Chapter 5 is based on the work of a jointly-authored publication entitled 'Quantum Secrecy in Thermal States' which, at the time of writing, had been accepted for publication in Journal of Physics B, and is available to view at arXiv.org, quant-ph, arXiv:1711.06592. This was authored by Elizabeth Newton, Anne Ghesquière, Freya L. Wilson, Benjamin T. H. Varcoe and Martin Moseley. The parts of this publication directly attributed to this author are limited to the description and figures relating to the practical experiment. The concept was a product of all of the above authors, and the theory and simulations predominantly carried out by A. Ghesquière.

This copy has been supplied on the understanding that it is copyright material and that no quotation from the thesis may be published without proper acknowledgement.

©2018 The University of Leeds and Elizabeth Newton

This thesis is dedicated to the Haddon through Knowle physics team.

Acknowledgements

My thanks to the entirety of the experimental quantum information department at the University of Leeds for their help and support. Especially Matthew Everitt and Freya Wilson, who were working on related projects and provided a lot of helpful discussion.

In addition, I would like to thank Timothy Newton and Ruth Newton who were willing to help with my maths on occasion, once they had stopped laughing. Also their unwavering moral support.

My thanks to Mohsen Razavi for his support and supervision, as well as very helpful discussions.

Finally, this project would not have been possible without my supervisor, Ben Varcoe, the king of story time.

Abstract

Quantum cryptography aims to provide secure communications, no matter how powerful or resourceful an eavesdropper is. As long as the eavesdropper is constrained by the laws of physics, she can never break the underlying cryptographic system. One of the key applications in quantum cryptography is key distribution. This involves the creation and sharing of the cryptographic key, the element of the message which provides secrecy between the sender and the recipient. There are two main forms of quantum key distribution. The first uses discrete variables, where information is encoded onto discrete modes of light, such as the polarisation of single photons. The second uses continuous variables, where information is encoded onto an infinite dimensional space, such as the quadratures of coherent states of light, which generally consist of several photons.

Here, an additional approach is examined. It falls under the category of continuous variable quantum key distribution, but instead of coherent states, such as those produced by lasers, it uses thermal states, produced by thermal sources of light. First, the experimental properties of thermal states are examined and compared to those of coherent states, then a simple key distribution model is constructed to test if thermal states are viable as a resource for secrecy. The initial results look promising.

When using continuous variables for key distribution, information is encoded by using randomly generated quadratures with a Gaussian distribution. Several methods for discretising these quadratures into key bits are examined to see if they have security implications associated with them. It is found that a poor choice of discretisation method can leak information to an eavesdropper. Once the quantum transmission is complete for key distribution, some classical error correction needs to be performed. A new eavesdropping technique for an eavesdropper to attack one of these forms of error correction, cascade, is described. We find that more attention needs to be paid to the classical parts of quantum key distribution in order for it to succeed.

Abbreviations

ADC	Analogue to Digital Converter
ASE	Amplified Spontaneous Emission
AWGN	Additive White Gaussian Noise
BB84	Bennett and Brassard 1984 quantum cryptography protocol
BS	Beam Splitter
CSV	Comma Separated Variable
CVQKD	Continuous Variable Quantum Key Distribution
DAQ	Data AcQuisition device
ECL	External (or Extended) Cavity Laser
EK91	Ekert 1991 quantum cryptography protocol
FFT	Fast Fourier Transform
FLFSR	Fibonacci Linear Feedback Shift Register
FWHM	Full Width at Half Maximum
GPS	Global Positioning System
LDPC	Low Density Parity Check
LED	Light Emitting Diode
MDRA	Multi Dimensional Reconciliation Algorithm
ND	Neutral Density
NIST	National Institute of Standards and Technology
OD	Optical Density
OSA	Optical Spectrum Analyser
PBS	Polarising Beam Splitter
PC	Personal Computer
POVM	Positive Operator Valued Measurement
QKD	Quantum Key Distribution
RMS	Root Mean Squared
SEC	Sliced Error Correction
SLED	Super LuminEscent Diode (Sometimes SLD)
SNR	Signal to Noise Ratio
TEC	ThermoElectric Cooling
USB	Universal Serial Bus
XOR	eXclusive OR logic gate

Contents

List of Figures

Chapter 1

Background

Quantum cryptography enables us to create perhaps one of the most powerful, and controversial, tools for this age; the ability to communicate over distance in a way which is unconditionally secure. No adversary, no matter how strong, how rich, how powerful, or how technologically advanced can intercept the content of those messages.

This is however currently a technology which, while maybe not still in its infancy, has reached, at furthest, those tricky adolescent years. Competing methods and protocols vie for attention, viable attacks pop up like acne, and mature classical cryptographic systems regard it with anything from exasperation to amused indulgence.

In this work, we make steps towards joining the throng of new quantum protocols, as well as investigating the security of some commonly used sub-methods.

The protocol we present is based on the possibility of exploiting properties of thermal states in order to enhance the secrecy and information rate of a continuous variable quantum key distribution scheme. In order to do this, there are a number of background concepts which we first need to explore. Chapter 1 covers sections on correlation functions and thermal light, information theory, and existing quantum cryptographic protocols.

This is followed by chapter 2, which delves into more detail for a few specific cryptographic protocols and how they relate to the one we propose. Further sections of chapter 2 begin to examine in detail the classical reconciliation stages necessary for tasks such as error correction.

In chapter 3, we lay out our new protocol, which uses thermal light to try and enhance the secrecy of the transmission.

Chapter 4 details the set-up and characterisation of the light source, which is used in a novel experiment detailed in chapter 5 to examine the possibility of using thermal states in quantum cryptography. We find that security is enhanced in some regions and lowered in others.

Chapter 6 focuses on the classical end of the protocol and is split into two sections. Section 6.1 provides a new examination of the security during the discretisation of Gaussian variables, and section 6.2 describes an innovative eavesdropper attack on cascade, a classical reconciliation method.

The conclusions of this work are presented in chapter 7.

It is important to note that we make no attempt to provide a full security proof of our new protocol, rather we use it as a prototype on which we can begin to examine the individual building blocks in some detail. It is hoped that one day it will form the basis of a rigorous protocol, built on the work presented here. It is vital however, that each individual step of the protocol is examined thoroughly as well as the interactions between the steps, and this is what we begin to attempt here.

1.1 Correlation functions and thermal light

1.1.1 Coherence and correlations

> ***"Just what is coherence? The answer typically depends on whom you ask!"***

- P. Meystre & M. Sargent III [1]

Coherence seems to be a tricky concept to define. You have after all, the linguistic meanings - the action of sticking together, a logical connection, consistency, agreement [2] - which while playing on a central theme (one might even say a coherent one), are not all that precise. Different fields appear to define coherence in subtly different ways, with the central theme being a process "characterised by the existence of a well-defined deterministic phase relationship" - [1].

Take for example a laser. Lasers can be spatially coherent and temporally coherent. They can have coherence times and coherence lengths. They can be made from coherent states, a coherent superposition of states. Many of the experimental techniques used with lasers can be described as coherent or incoherent. Yet many of these things are completely unrelated.

A spatially coherent beam has a fixed phase relationship between the electric fields across the width of the beam (perpendicular to the direction of travel). A temporally coherent one has the same fixed phase relationship, but in the direction of travel. The coherence time is the degree of time over which the temporal coherence is lost. The coherence length is the coherence time multiplied by the speed of light. Perhaps somewhat unintuitively given its name, it characterises the temporal rather than the spatial coherence.

Techniques described as coherent or incoherent usually refer to whether or not the method is phase sensitive.

Coherent states, on the other hand, refer to a quantum state of light described by Glauber [3]. They consist of a coherent superposition of Fock states (states with a well defined number of photons). In these coherent states, the beam is not necessarily spatially or temporally coherent.

1.1.2 First order coherence

We shall look here at coherence as used in quantum optics. This is nicely described by Glauber in multiple texts, most notably [4]. A summary of which is provided below.

We start with Young's double slit experiment[1], shown in figure 1.1. A single source of light is shone through two pinholes, small enough to avoid any diffraction effects, creating perfectly isotropic waves on the other side. This light then falls onto a screen where we measure its intensity.

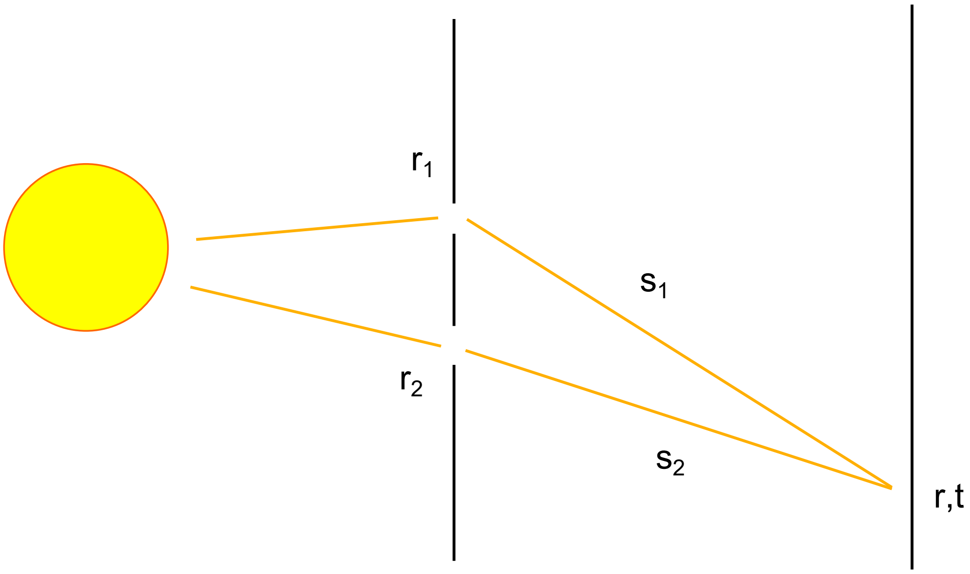


Figure 1.1: Young's double slit experiment.

We're interested in the field at point r, time t on the screen. This point lies at different distances s_1 and s_2 from the two pinholes r_1 and r_2, and is a superposition of the light from each. The times taken for these two different portions of the beam to reach point r are given by $t_1 = t - \frac{s_1}{c}$ and $t_2 = t - \frac{s_2}{c}$ respectively. The field is given as

$$E_r(t) = E(t_1) + E(t_2), \tag{1.1}$$

[1] An equivalent analysis performed by Loudon [5] uses a Michaelson interferometer.

and intensity found by averaging over a complete cycle:

$$\bar{I}(t) = \frac{1}{2}\epsilon_0 c|E_r(t)|^2 \tag{1.2}$$

$$= \frac{1}{2}\epsilon_0 c\{|E(t_1)|^2 + |E(t_2)|^2 + 2\Re[E^*(t_1)E(t_2)]\}. \tag{1.3}$$

The observation period is likely to be much longer than the period of the source, so we introduce the time averaged intensity

$$\langle\bar{I}(t)\rangle = \frac{1}{2}\epsilon_0 c\{\langle|E(t_1)|^2\rangle + \langle|E(t_2)|^2\rangle + 2\Re\langle E^*(t_1)E(t_2)\rangle\}. \tag{1.4}$$

The first two terms are independent, so cause no interference effects. All of the interference is caused by the second term, so this is called the first order correlation function. More formally,

$$G^{(1)} = \langle E^*(t_1)E(t_2)\rangle \tag{1.5}$$

$$= \frac{1}{T}\int_T dtE^*(t_1)E(t_2). \tag{1.6}$$

The first two terms of equation 1.4 give the contribution to the intensity from each pinhole. The final term is due to the fact that it is impossible to know which pinhole any particular photon came from.

This can be normalised to its more usual form

$$g^{(1)}(\tau) = \frac{\langle E^*(t)E(t+\tau)\rangle}{\langle E^*(t)E(t)\rangle}. \tag{1.7}$$

You can of course take $G^{(1)} = |G^{(1)}|e^{i\phi}$, to give

$$\langle\bar{I}(t)\rangle = \frac{1}{2}\epsilon_0 c\{\langle|E(t_1)|^2\rangle + \langle|E(t_2)|^2\rangle + 2\langle|E^*(t_1)E(t_2)|\rangle\cos\phi\}. \tag{1.8}$$

As you move the detector (r, t) up and down the screen, s_1, s_2 change, and along with them, t_1, t_2. As you do so, the intensity varies in a sinusoidal pattern, governed by that last term. If this vanishes, the sinusoidal fringes disappear, and the light from the two pinholes is described as uncorrelated, or incoherent.

When it is at a maximum, the fringes show the most contrast, and the light it correlated, or first order coherent[1]. This upper bound is given by the Schwarz inequality

$$|G^{(1)}(t_1, t_2)|^2 \leq G^{(1)}(t_1, t_1)G^{(1)}(t_2, t_2). \tag{1.9}$$

Two fields are coherent at points r_1, t_1 and r_2, t_2 if the Schwarz inequality holds. The entire field is coherent if it holds for all space-time points.

Thermal light, emitted from a black body (often also referred to as “chaotic” or “incoherent” light), obviously has different properties to the coherent light emitted from a laser. The question is, how different is the light created from these two sources? We can start with the obvious, classically observable, differences. Lasers produce light with very strong spatial and temporal coherence. Black bodies do not. This means that lasers can produce beams of light which do not diverge as they travel over distance. They can also produce monochromatic light, that of a single wavelength. The question is if, theoretically, you could filter the light emitted from a black body source until it had these same properties of the laser light, would you still be able to tell the difference between them? The spectra are identical, the first order $G^{(1)}$ correlations are identical, are there any other measures that could be used?

1.1.3 Second order coherence

It is possible to extend the correlation function into higher orders. So for this next one, we move from Young’s double slit experiment, to the Hanbury Brown Twiss interferometer.

Robert Hanbury Brown, and Richard Twiss were astronomers who developed a new interferometer. Until this point, if you wanted to measure the diameter of a discrete radio source, you had to use a Michelson interferometer [6], shown in figure 1.2.

These consisted of two antenna, spaced some distance apart along a baseline. A cable (or radio link over long distances) connected the two antenna along this

[1] First order coherence is sometimes referred to as optical coherence

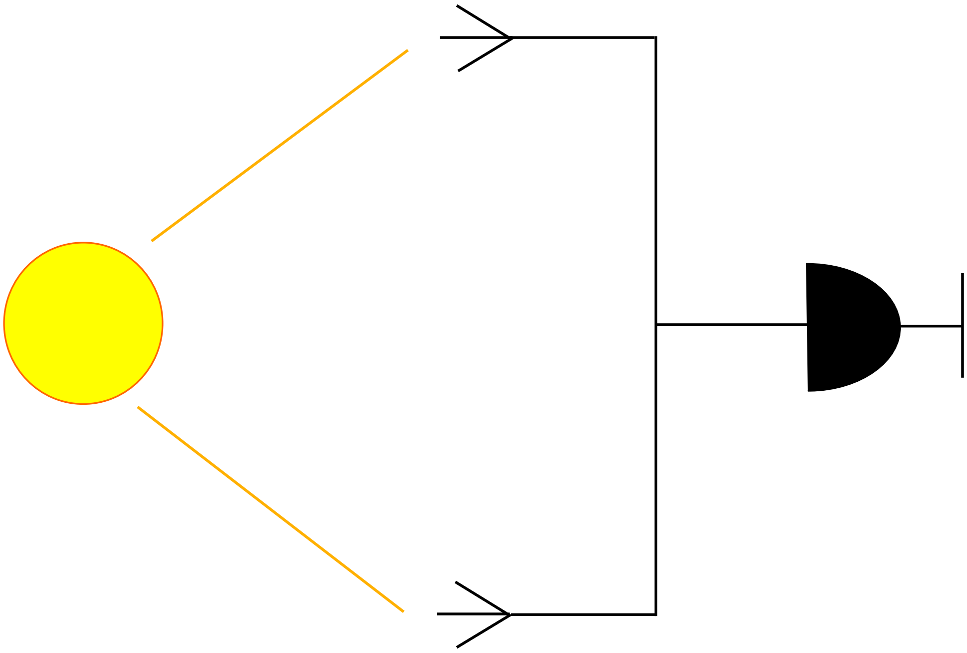


Figure 1.2: The Michelson interferometer.

baseline, with the centre of this cable connected to a receiver. The receiver measures the combined output of the two antenna. As a radio source moves across in front of the antenna, parallel to the baseline, an oscillation in output power is observed from the receiver. This is analogous to the interference fringes we saw in Young's double slit experiment, only here instead of moving the detector, we're moving the source. In this case, the wavefronts from each arm have the opportunity to interfere where the cables join before the detector, meaning that it is important to maintain the phase.

We are however now in the real world, where instead of a point source, you're using a star of finite size. This means that the power received at each arm is the integral of the energy distribution across the surface of the source. Using longer baselines allows you to resolve smaller angular diameter sources.

The limitations of this are immediately obvious. Stars frequently have quite small angular diameters, so astronomers very quickly found themselves requiring impractically long baselines. In addition to this, the phase of the signal needs to be preserved along the length of the baseline which is difficult in practice.

To overcome this, Hanbury Brown and Twiss developed a new interferometer which is very similar to the Michelson interferometer, but the signals are correlated after they're measured rather than before. This gets rid of the requirement to preserve the relative phase of the signals.

The basic set up for the Hanbury-Brown Twiss interferometer (figure 1.3) is, at first glance, very similar to that of the Michelson. However, the small differences end up, in fact, to be crucial. Here, the intensities of the signals are detected soon after measurement and then the intensities of each branch are correlated.

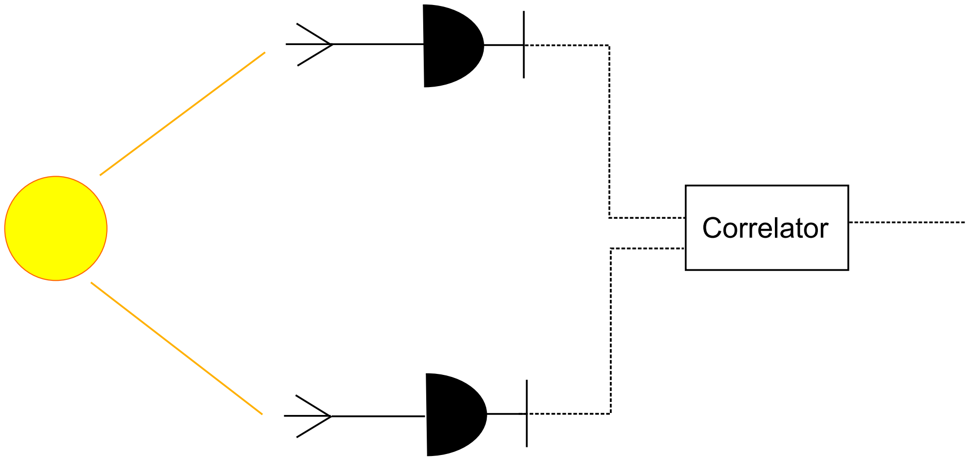


Figure 1.3: The Hanbury Brown Twiss interferometer.

In the Michelson interferometer (and Young's double slit experiment), the interference pattern is produced by the interference of the wavefronts. For this to happen, the phase needs to be preserved. In the Hanbury Brown Twiss system, all the phase information is discarded. Instead, the system relies on small intensity fluctuations in the incoming light, caused by the bunching of photons.

At first, there was some controversy [7], mainly due to the fact that at that time, the behaviour of light was far from well understood. It wasn't until 1963 that Glauber published a full framework for higher order photon correlations [8].

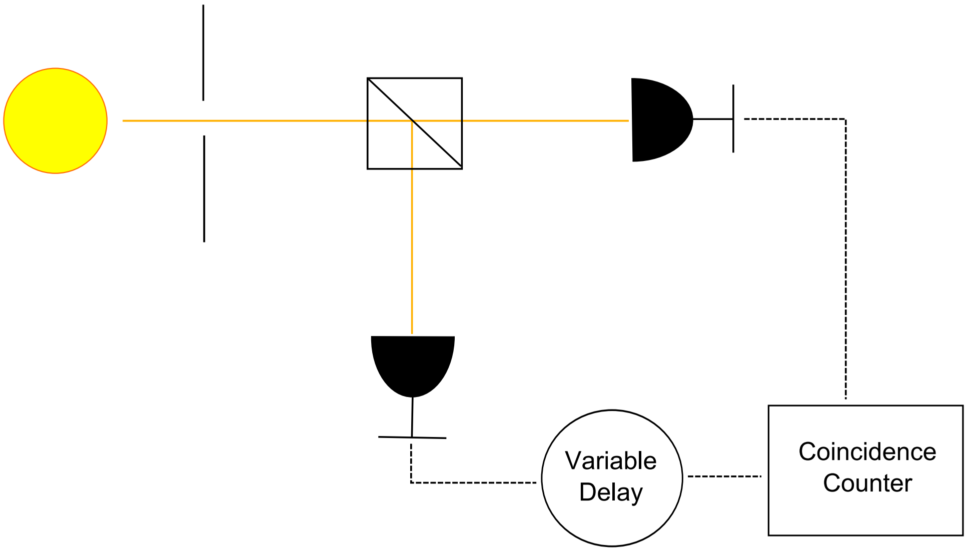


Figure 1.4: A general experimental set-up to examine second order coherence.

In figure 1.4, a more general form of the Hanbury Brown Twiss experiment, light is split between two detectors, and a variable time delay added to one arm, before the two arms are compared in a coincidence counter.

The beamsplitter allows us to create the same field in front of both detectors, (the other option being to put both detectors in the same place at the same time, which could be a little tricky experimentally)[1]. Adding the variable delay to one arm allows us to compare the same field to itself at different points in time. You then plot the coincidence rate, $G^{(2)}$ as a function of the time delay, τ.

If the photon counting events at the two detectors were statistically independent, then you would see a straight line at unity in this graph. However, if photons enter the system in pairs, or exhibit bunching, then the coincidence rate

[1] Humorous comment first given by Glauber [9] and far too good to miss.

can double around $\tau = 0$.

Lasers produce first order coherent light with statistically independent photons. The light from them exhibits no increase in coincidence rate, and no second order correlations. Thermal light however, produces bunched photons. This causes the increase in coincidence rate, and second order coherence.

We asked earlier how it could be possible to tell the difference between laser light and highly filtered black body light. This is how. Black body, or thermal, light has second order correlations, whereas coherent light from lasers does not.

For Gaussian states, it is possible to construct correlation functions to the n$^{\text{th}}$ order by expressing them as products of the first order correlations, and summing over all possible permutations of coordinates.

$$G^{(n)}(x_1...x_n, y_1...y_n) = \sum_P \prod_{j=1}^{n} G^{(1)}(x_j y_{P_j}), \tag{1.10}$$

which means that for the second order correlation,

$$\begin{aligned} G^{(2)}(x_1 x_2 x_2 x_1) &= \sum_P \prod_{j=1}^{n} G^{(1)}(x_j x_{P_j}) && (1.11)\\ &= G^{(1)}(x_1 x_2) G^{(1)}(x_2 x_1) + G^{(1)}(x_1 x_1) G^{(1)}(x_2 x_2) && (1.12)\\ &= G^{(1)}(x_1 x_1) G^{(1)}(x_2 x_2) + |G^{(1)}(x_1 x_2)|^2. && (1.13) \end{aligned}$$

In a first order coherent field, e.g. from a laser, the second term vanishes, leaving you only with the first. There would be no photon coincidences above the background, as there is no bunching of photons. You would have a $G^{(2)}$ of unity. However, in thermal light, the Gaussian distribution of field amplitudes gives rise to the second term. This is what is responsible for the Hanbury Brown Twiss effect.

In simple terms, in a thermal field, the amplitude is fluctuating all the time. This means that detectors have to be correlated. In a coherent field, the amplitude is very well defined, so there are no correlations.

A note on distance.

Before we start thinking about using these correlations for cryptography, we need to consider the practicalities. A common concern among the QKD community when involving correlations of any kind, is their longevity. This is a natural concern given the fragility of quantum entanglement.

There are three main points to address here. The first is that given that Hanbury Brown and Twiss used their original interferometer to measure the angular diameter of Sirius, it is clear that these second order correlations can survive the literally astronomical distances of space.

Second, is that the degree of correlation seems to depend on the channel losses. This is discussed further in chapter 4. The higher the losses, the lower the correlations. This makes sense as with high losses, the proportion of fluctuations will be lower, and thus lower correlations are observed.

Thirdly is an interesting point on the linewidth of the source. The linewidth of a light source is the full width at half maximum (FWHM) of its optical spectrum, a measure of how wide its wavelength spread is. This is related to the coherence time, τ_{coh} of the source, via

$$L_{coh} = c\tau_{coh} = \frac{c}{\Delta\nu} \approx \frac{\lambda_0^2}{\Delta\lambda}, \tag{1.14}$$

where $\Delta\nu$ is the linewidth in Hertz, $\Delta\lambda$ the linewidth in m, c the speed of light, λ_0 the centre wavelength, and L_{coh} the coherence length.

For correlations to be observed, both Alice and Bob need to measure the beam within the coherence time of the source. This means that the distance between their detectors and the beamsplitter has to be within the coherence length of each other. The total distance of the detectors from the beamsplitter is irrelevant, it is only important that the distances of each match to within one coherence length.

Typical laser linewidths of commercial telecommunications equipment can easily reach as low as 10 kHz, giving a coherence length of 30 km. This means that while it is necessary to remember that the detectors have to be similar distances from the beamsplitter, and to design systems around this, it is not in itself an obstacle to using the second order correlations available in thermal light as a source of QKD.

1.1.4 The detector correlation function

The second order correlation function can be written

$$G^{(2)} = \langle E^{-}(t)E^{-}(t+\tau)E^{+}(t)E^{-}(t+\tau)\rangle, \tag{1.15}$$

where E^{+} are the positive frequency terms and E^{-} the complex conjugate terms. This can then be normalised to

$$\begin{aligned} g^{(2)} &= \frac{\langle E^{-}(t)E^{-}(t+\tau)E^{+}(t)E^{-}(t+\tau)\rangle}{\langle E^{-}(t)E^{+}(t)\rangle\langle E^{-}(t+\tau)E^{+}(t+\tau)\rangle} && (1.16) \\ &= \frac{\langle I(t)I(t+\tau)\rangle}{\langle I(t)\rangle\langle I(t+\tau)\rangle}. && (1.17) \end{aligned}$$

However, in our experiment, we are not directly measuring the incident light, but rather the production of a photoelectric current in the detector. While it may seem that there is not much difference between the two, in practice, we can exploit small differences in the definition to our advantage.

In early experiments, such as that by Hanbury Brown and Twiss [10], measuring the actual $g^{(2)}$ correlations from individual photon counts, or incident intensities, was difficult for a number of reasons, mainly the speed of their detectors[1].

[1] For more information about detectors, see section 4.2.1.

Hanbury Brown and Twiss were only interested in the presence or absence of the $g^{(2)}$ correlations, so instead used a slightly different function which measured the correlations in the outputs of their photocathodes. This new correlation function (called $C(d)$, or $\Gamma(d) = \frac{C(d)}{C(0)}$ in the normalised form), contained elements to account for factors such as the amplifier gain, the efficiency of the photocathodes, bandwidth of the amplifiers, and the wavelength of the incident light.

An equivalent, and slightly more up to date, form is derived in detail in [11]. It is summarised as follows.

A photodiode produces a photocurrent J, which is made up of j current pulses k caused by photons, emitted at time t_j hitting the detection surface at time t.

$$J(t) = \sum_j k(t - t_j). \tag{1.18}$$

The expectation value for a fixed number of photons, n, over time interval T is

$$\langle J_{n,T}(t)\rangle_n = \sum_{j=1}^{n} \int_{-\frac{T}{2}}^{\frac{T}{2}} k(t - t_j)\frac{dt_j}{T}, \tag{1.19}$$

where t_j is the time at which a photoelectron j is emitted within the time dt_j.

$$\langle J_{n,T}(t)\rangle_n = \frac{1}{T}\sum_{j=1}^{n} k(t_j)dt_j \tag{1.20}$$

$$\approx \frac{1}{T}\sum_{j=1}^{n} Q_j, \tag{1.21}$$

where $Q_j = \int_{-\infty}^{\infty} k(t_j)dt_j$, the total charge caused by a single current pulse. Averaging over the number of photons gives

$$\langle J(t)\rangle = \frac{\langle n\rangle Q}{T} = \eta\langle I\rangle Q, \tag{1.22}$$

which is simply the average number of current pulses, and the charge from each pulse, divided by the time interval. $\frac{\langle n\rangle}{T} = \langle I\rangle$, is the average incident light intensity, and η the detector efficiency.

From here we can calculate the auto- and cross- correlation functions. The auto correlation function is a measure of the correlation between the output of a detector at times t and $t+\tau$. The cross correlation is the same as this, but here we're looking at the outputs of two different detectors.

The cross correlation function is relatively simple,

$$\langle J_{1_{n,t}}(t)J_{2_{n,t}}(t+\tau)\rangle_n = \left\langle \sum_{i=1}^{n}\sum_{j=1}^{n} k(t-t_{1_i})k(t-t_{2_j}+\tau)\right\rangle_n \tag{1.23}$$

$$\langle \Delta J_1(t)\Delta J_2(t+\tau)\rangle = \eta_1\eta_2 \iint_{-\infty}^{\infty} k(t')k(t'')\langle \Delta I_1(t)\Delta I_2(t+t'-t''+\tau)\rangle dt'dt'' \tag{1.24}$$

The auto correlation, on the other hand is a bit more interesting.

$$\langle J_{n,t}(t)J_{n,t}(t+\tau)\rangle_n = \left\langle \sum_{i=1}^{n}\sum_{j=1}^{n} k(t-t_i)k(t-t_j+\tau)\right\rangle_n \tag{1.25}$$

Unlike the cross correlation, this can be split into two parts, one for the detection of a single photon, and one for the joint detection of a bunched pair.

$$\begin{aligned}\langle J_{n,t}(t)J_{n,t}(t+\tau)\rangle_n &= \left\langle \sum_{j=1}^{n} k(t-t_j)k(t-t_j+\tau)\right\rangle \\ &+ \left\langle \sum^{n}\sum_{i\neq j}^{n} k(t-t_i)k(t-t_j+\tau)\right\rangle\end{aligned} \tag{1.26}$$

After some algebra which we will gloss over for convenience[1],

$$\begin{aligned}\langle \Delta J(t)\Delta J(t+\tau)\rangle &= \eta\,\langle I\rangle \int_{-\infty}^{\infty} k(t')\,k(t'+\tau)\,dt' \\ &+ \eta^2 \iint_{-\infty}^{\infty} k(t')\,k(t'')\,\langle I(t)\,I(t+t'-t''+\tau)\rangle\,dt'dt''\end{aligned} \tag{1.27}$$

[1] again, details can be found in [11]

Giving,

$$\begin{aligned}\langle \Delta J(t)\Delta J(t+\tau)\rangle = \eta\,\langle I\rangle \int_{-\infty}^{\infty} k(t')\,k(t'+\tau)\,dt' \\ + \eta^2 \iint_{-\infty}^{\infty} k(t')\,k(t'')\,\langle \Delta I(t)\,\Delta I(t+t'-t''+\tau)\rangle\,dt'dt''.\end{aligned} \tag{1.28}$$

These are the cross correlation (equation 1.24) and auto correlation (equation 1.28) functions in terms of the incident intensity, and detector properties. Due to the fact that the response times of our detectors are considerably longer than the coherence time of the light[1], we can simplify these equations somewhat.

When $\tau = 0$, there is no time delay between the measurement of the detector(s), then it is reasonably clear that $\langle \Delta I(t)\Delta I(t+\tau)\rangle = \langle (\Delta I)^2\rangle$ for auto correlations, and $\langle \Delta I_1(t)\Delta I_2(t+\tau)\rangle = \langle \Delta I_1\Delta I_2\rangle$ for cross correlations.

However, when $\tau \neq 0$, if the reaction time of the detector T_R is significantly longer than the timescale of the correlations (the coherence time τ_c), then the intensity correlation in the integral can be written

$$\langle \Delta I(t)\Delta I(t+\tau)\rangle = \langle (\Delta I)^2\rangle \tau_c \delta(\tau) \tag{1.29}$$

$$\langle \Delta I_1(t)\Delta I_2(t+\tau)\rangle = \langle \Delta I_1\Delta I_2\rangle \tau_c \delta(\tau), \tag{1.30}$$

which means the auto- and cross- correlations become

$$\begin{aligned}\langle \Delta J(t)\Delta J(t+\tau)\rangle = \eta\langle I\rangle \int_{-\infty}^{\infty} k(t')k(t'+\tau)dt' \\ + \eta^2\langle (\Delta I)^2\rangle \tau_c \int_{-\infty}^{\infty} k(t')k(t'+\tau)dt'\end{aligned} \tag{1.31}$$

$$\langle \Delta J_1(t)\Delta J_2(t+\tau)\rangle = \eta_1\eta_2\langle \Delta I_1\Delta I_2\rangle \tau_c \int_{-\infty}^{\infty} k(t')k(t'+\tau)dt' \tag{1.32}$$

respectively.

[1] See section 4.2.3 for a detailed analysis.

For thermal light,

$$\langle (\Delta I)^2 \rangle = \frac{1}{2}\langle I \rangle^2 \tag{1.33}$$

$$\langle \Delta I_1 \Delta I_2 \rangle = \frac{1}{2}\langle I_1 \rangle \langle I_2 \rangle |\gamma_{1,2}|^2, \tag{1.34}$$

where $|\gamma_{1,2}|$ is the equal time degree of coherence. So these become

$$\langle \Delta J(t) \Delta J(t+\tau) \rangle = [\eta \langle I \rangle + \frac{1}{2}\eta^2 \langle I^2 \rangle \tau_c] \int_{-\infty}^{\infty} k(t')k(t'+\tau)dt' \tag{1.35}$$

$$\langle \Delta J_1(t) \Delta J_2(t+\tau) \rangle = \frac{1}{2}\eta_1\eta_2 \langle \Delta I_1 \rangle \langle \Delta I_2 \rangle |\gamma_{1,2}|^2 \tau_c \int_{-\infty}^{\infty} k(t')k(t'+\tau)dt'. \tag{1.36}$$

The normalised correlation function, in analogy to the $g^{(2)}$ can be written

$$C(\tau) = \frac{\langle \Delta J_1(t) \Delta J_2(t+\tau) \rangle}{\sqrt{\langle [\Delta J_1(t)]^2 \rangle}\sqrt{\langle [\Delta J_2(t+\tau)]^2 \rangle}} \tag{1.37}$$

and then

$$C(\tau) = \frac{\frac{1}{2}\sqrt{\eta_1\eta_2 \langle I_1 \rangle \langle I_2 \rangle}\tau_c |\gamma_{1,2}|^2}{\sqrt{1+\frac{1}{2}\eta_1 \langle I_1 \rangle \tau_c}\sqrt{1+\frac{1}{2}\eta_2 \langle I_2 \rangle \tau_c}}. \tag{1.38}$$

This provides a measure of the correlations between the two detectors, but is not the same as the $g^{(2)}$ function. It is helpful in proving the existence of second order correlations, but cannot be used in the same way as the $g^{(2)}$ function to characterise them. Indeed, it is generally plotted as $\frac{C(\tau)}{C(0)}$ to make it look neater.

1.1.5 Linewidth broadening

In the previous section, we saw that the detector response times had an effect on the correlation function between the two detectors. Logically, this must also effect the $g^{(2)}$ correlation.

The following analysis is based heavily on that of Loudon [5].

There are a number of mechanisms in the production of the light which can contribute to broadening the linewidth (the range of frequencies). These are mainly, collision, radiation and Doppler broadening. Following the example in

[5], we look at the effect of collision broadening, with its characteristic spiky fractal signature.

An excited atom will radiate light steadily at a frequency of ω_0 until it suffers a collision. We don't worry about what happens during the collision, because that doesn't last very long, but after the collision the atom continues radiating light. This post-collision light is identical in every way to the pre-collision light except it has undergone a random phase change.

The field amplitude of the wave emitted by the atom is

$$E(t) = E_0 \exp^{(-i\omega_0 t + i\phi(t))}, \tag{1.39}$$

where $\phi(t)$ is the phase. This changes after each collision.

The wave emitted by the source as a whole is the sum of the contributions from each atom:

$$\begin{aligned} E(t) &= E_1(t) + E_2(t) + \cdots + E_v(t) && (1.40) \\ &= E_0 \exp^{(-i\omega_0 t)} \exp^{(i\phi_1(t))} + \exp^{(i\phi_2(t))} + \cdots + \exp^{(i\phi_v(t))} && (1.41) \\ &= E_0 \exp^{(-i\omega_0 t)} a(t) \exp^{(i\phi(t))}, && (1.42) \end{aligned}$$

where v is the number of atoms.

The last line of equation 1.40 is due to the fact that the different phases of all the atoms can be viewed as a random walk, so are now represented by an amplitude $a(t)$ and phase $\phi(t)$ which change with time.

In effect, you now have a carrier wave of frequency ω_0 with added random phase and amplitude modulation. The Fourier decomposition of this modulated wave contains frequencies spread around ω_0, which gives the broadened linewidth.

We know from section 1.1.2 that the first order coherence can be written

$$\langle E^*(t)E(t+\tau)\rangle = \frac{1}{T}\int_T dtE^*E(t+\tau), \tag{1.43}$$

which we can expand in the collision broadened case to

$$\begin{aligned}\langle E^*(t)E(t+\tau)\rangle = {} & E_0^2 \exp^{(-i\omega_0\tau)} \\ & \times \langle \exp^{(-i\phi_1(t))} + \exp^{(-i\phi_2(t))} + \cdots + \exp^{(-i\phi_v(t))} \\ & \times \exp^{(i\phi_1(t+\tau))} + \exp^{(i\phi_2(t+\tau))} + \cdots + \exp^{(i\phi_v(t+\tau))}\rangle \end{aligned} \tag{1.44}$$

$$= E_0^2 \exp^{(-i\omega_0\tau)} \sum_j^v \langle \exp^{i[\phi_j(t+\tau)-\phi_j(t)]}\rangle \tag{1.45}$$

$$= v\langle E_j^*(t)E_j(t+\tau)\rangle. \tag{1.46}$$

The phases from different atoms have different random values, so give no overall contribution. This means that you only need to consider terms from the same atom. As all atoms contribute equally, this can be simplified to the single atom contribution function.

Because the phase of each atom changes to a different random value after each collision, the average contribution to the correlation of the whole field is zero. The correlation function for a single atom has to be proportional to the atom avoiding collisions for time periods greater than τ,

$$p(\tau)d\tau = \frac{1}{\tau_c}\exp^{-\frac{\tau}{\tau_c}} d\tau, \tag{1.47}$$

where $p(\tau)d\tau$ is the probability of free flight between τ and $d\tau$ seconds, and τ_c the coherence time.

From equation 1.44,

$$\langle E_j^*(t)E_j(t+\tau)\rangle = E_0^2 \exp^{(-i\omega_0\tau)} \langle \exp^{i(\phi_j(t+\tau)-\phi_j(t))}\rangle \tag{1.48}$$

$$= E_0^2 \exp^{(-i\omega_0\tau)} \int_\tau^\infty d\tau' p(\tau') \tag{1.49}$$

$$= E_0^2 \exp^{(-i\omega_0\tau-(\frac{\tau}{\tau_c}))}, \tag{1.50}$$

So the correlation function is

$$\langle E^{*}(t)E(t+\tau)\rangle = vE_0^2 \exp^{(-i\omega_0\tau - \frac{\tau}{\tau_c})}, \tag{1.51}$$

and once normalised,

$$g^{(1)}(\tau) = \exp^{(-i\omega_0\tau - \frac{|\tau|}{\tau_c})}, \tag{1.52}$$

For Doppler broadening (easily distinguishable from collision broadening by the smooth outline when plotting the intensity as a function of time), you have a $g^{(1)}$ of

$$g^{(1)}(\tau) = \exp^{(-i\omega_0\tau - \frac{\pi}{2}(\frac{\tau}{\tau_c})^2)}. \tag{1.53}$$

However, because all our data is not smooth, we don't concern ourselves with this.

It is possible to follow a similar procedure for the second order correlations [5], however here, you also have to take into account not only correlations between atoms and themselves, but also pairs of different atoms. In fact, the correlations between pairs of atoms dominate through sheer numbers. This produces the result,

$$g^{(2)}(\tau) = 1 + |g^{(1)}(\tau)|^2. \tag{1.54}$$

So the $g^{(2)}$ correlation for the Lorentzian frequency spectrum (collision and radiation broadening) is

$$g^{(2)}(\tau) = 1 + \exp^{(-2\frac{|\tau|}{\tau_c})}. \tag{1.55}$$

1.1.6 Detector integration times and second order correlations

This new definition of $g^{(2)}$ is useful as it allows us to calculate the effect on the measured height of the $g^{(2)}$ correlations due to the response time of the detector.

If the detector has an integration time T_R[11], then

$$g^{(2)} = 1 + \frac{1}{T_R^2} \iint_0^{T_R} \exp^{[-2(\frac{t_4 - t_3}{\tau_c})]} dt_4 dt_3 \tag{1.56}$$

$$= 1 + \frac{\tau_c^2}{2T_R^2} \Big[\exp^{\frac{-2T_R}{\tau_c}} - 1 + \frac{2T_R}{\tau_c} \Big]. \tag{1.57}$$

We will see later (section 4.2.3) that for slow detectors, or light with short coherence times, this can make a huge impact on the expected height of the $g^{(2)}(0)$ peak. In fact, you can see here that for $T_R >> \tau_c$, $g^{(2)} \to \frac{\tau_c}{T_R}$.

1.2 Information Theory

Information can be a difficult thing to try and conceptualise.

Entropy (H or S), also known as the Shannon entropy, to distinguish it from statistical mechanics, is a measure of the amount of disorder in a system. It can additionally be thought of as the expected (or average) amount of information in a message. Information can be thought of as how surprising something is. Something very surprising contains a lot of information, while something that isn't surprising (i.e. that you know already) contains little information.

Take for example, a coin toss.

If the coin is fair ($p(\text{heads}) = 0.5$, $p(\text{tails}) = 0.5$)[1] then this contains the most entropy as you cannot know beforehand what the outcome will be. The outcome will contain a lot of information. If however, the coin is weighted so that $p(\text{heads}) = 0.8$ and $p(\text{tails}) = 0.2$, then this will have a much smaller entropy (the outcome will contain less information) as you have a reasonable idea of what each outcome will be beforehand. A coin which always lands heads will have the minimum entropy.

[1] where $p(\text{heads})$ is the probability of getting a heads

This can be written

$$H = -\sum_i p_i \log_2 p_i, \tag{1.58}$$

where H is the entropy, i each possible outcome (eg. heads or tails), and p_i the probability of that outcome.

So for the 50:50 coin, $H = -2(0.5 \times log2(0.5)) = 1$, for the 80:20 coin, $H = -(0.8 \times log2(0.8) + 0.2 \times log2(0.2)) = 0.7$, and for a 100:0 coin, $H = -(1 \times log2(1)) = 0$.

Here we've used log base 2. This gives an answer in units of bits. If you use a natural log, the answer is in units of nats. There are a variety of different units for different log bases. Bits are generally the most natural unit to use as many systems have only two states, for example, physical bits in computers, flipping coins etc. If you were rolling a dice instead, with 6 possible states, you may want to use log base 6. However, because it is perfectly easy to convert between the different units, bits are the most commonly used.

1.2.1 Mutual information

Mutual information is a measure of how much information is shared between two parties. In this example we will call them Alice (A) and Bob (B) in line with the convention for cryptography. Mutual information can also be thought of as how much information you can tell about something by knowing something else. This is described by the equation

$$I(A;B) = -\sum_{a \in A} \sum_{b \in B} p(a,b) \log_2 \Big(\frac{p(a,b)}{p(a)p(b)} \Big). \tag{1.59}$$

So the information shared between two things, A and B, is written $I(A;B)$. a are the individual outcomes of A, and b the outcomes of B. $p(a,b)$ is the joint probability distribution of a and b, (how probable it is for both outcomes a and b to occur) and $p(a)$ and $p(b)$ are the marginal probability density functions (how

Alice	0	0	0	1	1	0	1	0
Bob	0	0	1	1	1	0	1	0

Table 1.1: Two bit strings for Alice and Bob

probable it is for each of a and b to occur).

If Alice has a string of bits: $A = 0\ 0\ 0\ 1\ 1\ 0\ 1\ 0$, and Bob has a string of bits on bit different, $B = 0\ 0\ 1\ 1\ 1\ 0\ 1\ 0$, to calculate the mutual information, you need to find $p(a, b)$, $p(a)$ and $p(b)$ for each circumstance. This is best described using a table (table 1.1).

The first possible circumstance is that both Alice and Bob have a 0. This happens four times from a possible eight, so $p(a = 0, b = 0) = \frac{4}{8} = 0.5$. The second possible circumstance is that Alice has a 1, and Bob a 0. This never happens, so $p(a = 1, b = 0) = 0$. Thirdly, Alice has 0 and Bob has 1, gives $p(a = 0, b = 1) = \frac{1}{8} = 0.125$, and lastly, where both have a 1, $p(a = 1, b = 1) = \frac{3}{8} = 0.375$. These should add to one, as there is always some combination of Alice and Bob.

We also need to know, the marginal probabilities, $p(a = 0) = \frac{5}{8} = 0.625$, $p(a = 1) = \frac{3}{8} = 0.375$, $p(b = 0) = \frac{4}{8} = 0.5$, and $p(b = 1) = \frac{4}{8} = 0.5$, these are simply counting the ones and zeroes for each of Alice and Bob.

Then it is a simple matter of summing each component.

$$\begin{aligned} I(A, B) = -\Big(&p(a=0,b=0)\log_2\left[\frac{p(a=0,b=0)}{p(a=0)p(b=0)}\right] \\ &+p(a=0,b=1)\log_2\left[\frac{p(a=0,b=1)}{p(a=0)p(b=1)}\right] \\ &+p(a=1,b=0)\log_2\left[\frac{p(a=1,b=0)}{p(a=1)p(b=0)}\right] \\ &+p(a=1,b=1)\log_2\left[\frac{p(a=1,b=1)}{p(a=1)p(b=1)}\right]\Big). \end{aligned} \tag{1.60}$$

Essentially, the mutual information is how similar Alice and Bob are. Venn diagrams are a useful way to illustrate concepts in information theory.

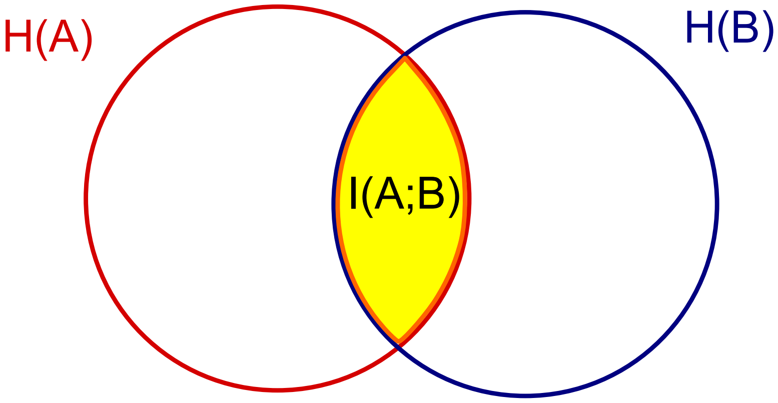


Figure 1.5: A Venn diagram displaying the mutual information between A and B

Figure 1.5 shows two circles. The red one is all of Alice's information, the entropy of which is shown as $H(A)$. The blue circle is Bob's. The overlapping sections are all the information which they both share, the mutual information $I(A;B)$.

1.2.2 Conditional entropy

There are two more sections of this diagram, which are as yet unlabelled. These are the the information which Alice has that Bob doesn't, and similarly the information Alice knows and Bob doesn't. These are called the conditional entropy (figure 1.6), and can also be thought about as the information you can know about Alice given what you know about Bob. It is written $H(A|B)$, and is found via

$$H(A|B) = -\sum_{a\in A}\sum_{b\in B} p(a,b)\log_2\left(\frac{p(a,b)}{p(b)}\right). \tag{1.61}$$

This is similar to the mutual information, but without the dependence on $p(a)$.

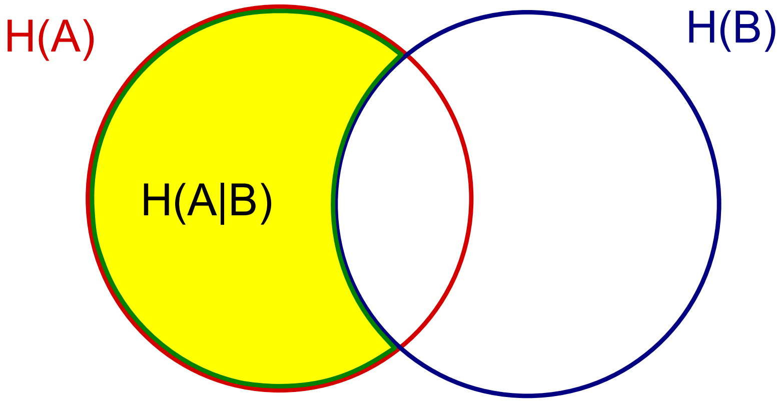


Figure 1.6: The conditional entropy

It is important to note that while $I(A;B) = I(B;A)$, $H(A|B)$ is not generally equal to $H(B|A)$.

1.2.3 Joint Entropy

Joint entropy is the whole area of the graph. It is all the information known by Alice and Bob combined, shown in figure 1.7.

It is written

$$H(A,B) = -\sum_{a \in A} \sum_{b \in B} p(a,b) \log_2(p(a,b)), \tag{1.62}$$

and it has no dependence on $p(a)$ or $p(b)$.

1.2.4 Conditional Mutual Information

Now we can introduce a third player to this system, often in cryptographic contexts this is used as an eavesdropper called Eve. In most communications systems, Alice and Bob aim to maximise their mutual information ($I(A;B)$), the things which they both know. In a cryptography scenario, they also want to minimise the information known by the eavesdropper. This generally involves maximising the conditional mutual information, $I(A;B|E)$.

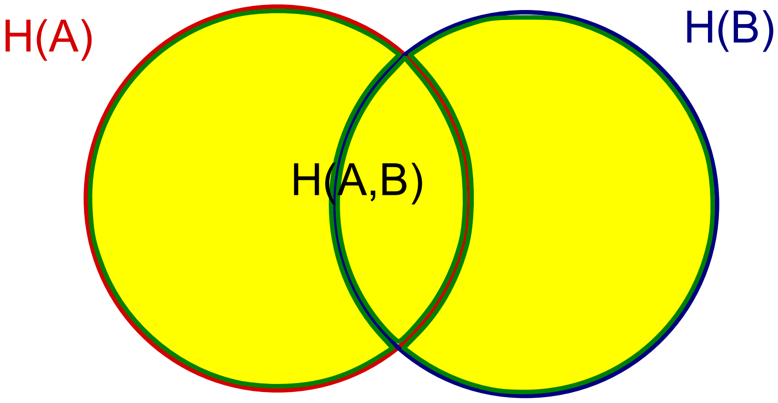


Figure 1.7: The joint entropy

This is information which Alice and Bob share, but Eve doesn't, and is written

$$I(A;B|E) = -\sum_a \sum_b \sum_e p(a,b,e) \log_2 \left(\frac{p(a,b,e)p(e)}{p(a,e)p(b,e)} \right). \tag{1.63}$$

The conditional mutual information is shown in figure 1.8.

1.2.5 Multivariate mutual information

The final area we look at is the multivariate mutual information. This is the mutual information between all three parties, Alice, Bob and Eve and marks the centre of the Venn diagram (figure 1.9). This can be written as $I(A;B;E)$, but is also frequently called R. It is found by subtracting the relevant conditional mutual information from the mutual information

$$I(A;B;E) = I(A;B) - I(A;B|E). \tag{1.64}$$

1.2.6 Information in practice

There are a number of mathematical methods for using these quantities. These include Bayes' rule,

$$H(A|B) = H(B|A) + H(A) - H(B), \tag{1.65}$$

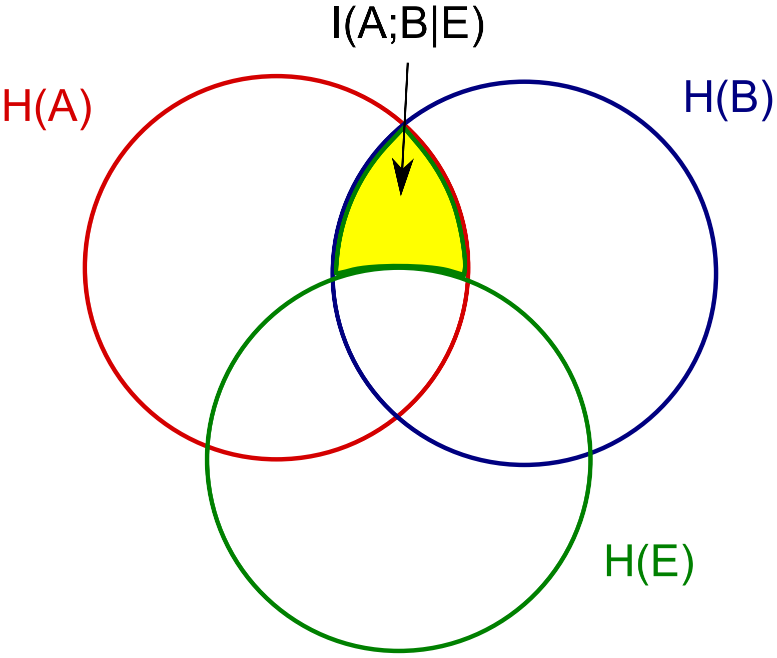


Figure 1.8: The conditional mutual information

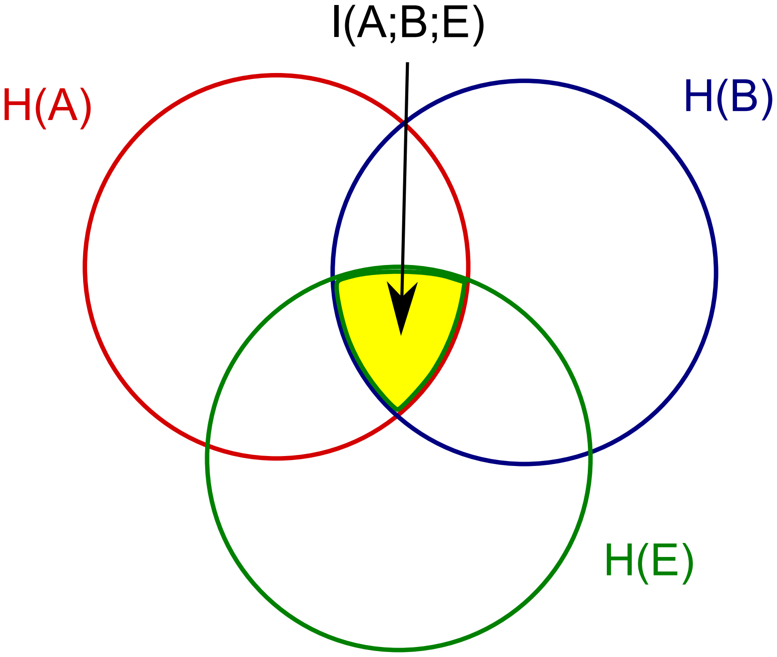


Figure 1.9: The multivariate mutual information

and the chain rule,

$$H(A|B) = H(A, B) - H(B), \tag{1.66}$$

amongst others. These are important for insuring mathematical rigour, but for general visualisation of these concepts and how to work with them, Venn diagrams are invaluable.

1.3 Cryptography

The history of cryptography can be traced back thousands of years, across languages, cultures and continents[1]. However, throughout all this time, there has only ever been one cypher which has been mathematically proven to be unconditionally secure.

1.3.1 Unconditional security

The Vernam cypher[2] is both supremely powerful and astonishingly simple. To use it, you fist change your message into binary, and then XOR (figure 1.10) this with a random sequence of zeroes and ones known as a key. As long as only the intended recipient of the message also knows this key, then the message is secure[17]. To decode the message, the recipient simply XOR's the message with the key to obtain the original plain text (figure 1.11).

The one-time pad [12, 13, 17] is a variation of the Vernam cypher, with much stricter conditions on the key[3]. The key has to be the same length as the message, random, with an equal distribution of 0's and 1's, and with no part of it repeated. Each of these cases is to prevent the eavesdropper working out, or reliably guessing, sections of the key. For example, if it's not sufficiently random then this increases her chances of correctly guessing the next bit. If there are more 1's than 0's then this increases the chances of the next bit being a 1 and

[1] Excellent introductions to the subject can be found in [12, 13], and articles on specific parts of interest are easy to come by, e.g. [14, 15]. [16] demonstrates how to construct a replica enigma machine from a pringles tube.

[2] Probably first discovered by Miller in the 1800's.

[3] Although the two terms are often used interchangeably.

Input		Output
0	0	0
0	1	1
1	0	1
1	1	0

Figure 1.10: The exclusive OR (XOR) table. The output is one if and only if one of the input values is one.

Plain text	0	0	0	0	1	1	1	1	0	1	0	1
Key	0	1	1	0	1	0	0	1	0	0	1	1
Message	0	1	1	0	0	1	1	0	0	1	1	0

Figure 1.11: An example of how a secret message can be constructed by XORing some binary text with a random key.

thus makes it easier for Eve to guess correctly. The key cannot be repeated for two reasons. Firstly, it makes subsequent sections of the key easier to guess as they are now predictable, and secondly, you are left with the same vulnerabilities as the Vigenère Cypher, whereby the repetition length of the key can be found and from this the message easily deciphered.

Once your binary key has been XORed with your message, it is unconditionally secure [17]. Only those who are in possession of the key can decipher it. This remains the only cryptographic method which remains secure no matter how much computing power Eve has at her disposal. The only difficulty is how to create and distribute the key. For this, we turn to quantum key distribution (QKD).

1.3.2 The Quantum Revolution

No discovery occurs in isolation. There will always be a cocktail of factors involved in its conception. The most important of which are the pressure points requiring its creation.

1.3.3 Quantum computers - The ultimate hackers

The field of quantum computing [18–20] was developed by a group of people (amongst them, Feynman [21], Deutsch [22], Benioff [23] and Manin [24]) in the 1980's. They intended to design, and one day create, a computer which instead of using binary bits (which take the absolute values 0 or 1), would instead run using quantum bits (or qubits) which can exist in a superposition of the states 0 and 1. To take full advantage of this, a new set of algorithms were designed. So far, a surprisingly small number of algorithms have been proposed [25], the most famous of which are Shor's and Grover's algorithms.

Grover's algorithm [26] allows you to efficiently search an un-ordered list, and Shor's algorithm [27] is very, very good at factorising. Unfortunately, a large part of cryptography in use today relies on the computational difficulty of factorising large numbers.

Luckily for the cryptography community, the technology required to build working quantum computers is still in its infancy (although growing up rapidly). There is a lot of argument as to the best architectures to use, and how to implement techniques such as error correction [18, 19]. Interestingly though, the technology seems to have moved to a point where the main advancements are now coming from industry rather than pure academia. There are some well-known technology giants making serious headway, including IBM [28], Microsoft [29], Google [30] and Intel [31], as well as a host of start-up companies such as Quantum Circuits [32], Ion Q [33], Rigetti Computing [34] and of course the controversial D-wave [35]. The main issue facing the field as a whole (apart from the disagreements over implementation) is scalability. While single qubits can be relatively easy to make, scaling them up to a workable number is significantly more difficult. The leaders in the field at time of writing are IBM with fifty [36].

Should this rapid progress be alarming to the cryptography community?

1.3.4 Classical cryptography and the quantum eavesdropper

In this digital age, vast numbers of different cryptography protocols are in use. Every time you sign into something on the web, spend money online, read a kindle book or even watch a DVD, cryptography is at work protecting data.

Cryptographic systems can be split into three main types - unkeyed, secret key, and public key.

Unkeyed systems are not really relevant to us, as the main purpose of quantum cryptography is key distribution. They are however mentioned here briefly, as they will crop up again later. The main type of unkeyed encryption system is a cryptographic hash function. Hash functions are used throughout computing as they have the remarkably useful property of being able to take a variable length input and turn it into an output of a fixed length. The output acts as an (almost[1]) unique identifier for the input. It is almost impossible to construct the input from the output, making them useful tools for cryptography. Hash functions can be designed to perform well at particular aspects of their use, such as minimising collisions, and some are designed especially for cryptographic applications. The more well-known of these are MD5 [37] and SHA-1 [38], although both of these have since been succeeded by improved versions and are no longer considered secure.

Secret key (or symmetric) encryption systems are the ones which we are most interested in. In these systems, both the sender and receiver (Alice and Bob) use the same key to encrypt and decrypt the messages. The key must remain secret from any outside party or eavesdropper. Examples include, AES [39], DES [40] and of course, the one-time pad. However, of these, only the first two are used in

[1] The cases where two different inputs produce the same output are called collisions.

practice. These practical cryptography systems are only conditionally or computationally secure. Unconditionally secure protocols, such as the one-time pad [17], need keys which are too long and bulky to be used in a practical environment [41].

The final type of encryption system is the public key (or asymmetric) cryptosystem. Here the receiver has a private key, known only to them. From this, they produce a public key which they release into the wild. Anyone can then encrypt data using the public key, but only the receiver with their private key can decrypt it. The most famous of these schemes is of course RSA [42]. These are the systems which tend to rely on the computational difficulty of factorising large numbers.

There is a lot of debate as to which forms of cryptosystem are vulnerable to quantum computers and to what extent. It is generally agreed that public key cryptosystems are the most vulnerable, due to their reliance on the computational difficulty of factorising large numbers, which becomes significantly less difficult with a working quantum computer. However, there are more and more papers such as [43] being published, promising that quantum computers could have the capability to hack symmetric key cryptosystems.

Is it time to panic?

The advice from the National Institute of Standards and Technology (NIST) is to not panic [44]. The current estimates are that it will take 10-20 years to make a sufficiently functioning quantum computer, but only 5-20 years to find a new encryption scheme. At the time of writing, NIST are collecting ideas, data and proposals, in order to create a set of standards for post-quantum cryptography. The main problem they are facing is that anything they implement has to not only be quantum secure, but also better than the current standard classically. There is no point going backwards mathematically, and forwards quantumly, because due to the expense of building a quantum computer, mathematics is by far the bigger threat with the worst consequences.

1.3.5 Classical solutions

There are currently three main classical encryption systems which will be quantum secure. They also provide sufficient mathematical security, and have survived years of testing. These are latticed based, code based, and multi-variate cryptography. They are not described here, but information on them can be found in [45, 46]. Each of these is interesting in its own right, and it will be interesting to see which (or how many) of these will become widespread.

It is highly unlikely that any quantum scheme will be included in NIST's evaluation. This is due to a number of factors, mainly the infancy of the field. The three classical post-quantum schemes mentioned above have all been around for about 20 years and have been attacked, tested and refined for the length of that time by experienced mathematicians and cryptographers. Quantum cryptography has yet to produce a small number of (or even a single) extremely well-studied protocol which has the trust and agreement of a large part of the quantum community. Even once this has been achieved, the protocol in question will need to go through the whole testing process again, by the classical community before it will be accepted. Then, the protocol will have to be not only more secure against both quantum, and mathematical attacks than its classical counterparts, but also more efficient and cost effective. The security should not be a problem, as quantum cryptography is designed to be unconditionally secure. The cost of such a system will decrease with time, but the efficiency is the main trade-off for the increased security.

Quantum cryptography is unlikely to be much use in helping security once quantum computers come into their own. There are a number of reasons for this. Firstly, maturity[1]. Quantum cryptography is not yet sufficiently advanced to produce a system everyone is happy enough with before the quantum computers become effective. Secondly, they solve the wrong problem. Public key encryption

[1] Although interestingly, the first protocol developed for quantum cryptography was published in 1984, more than ten years before Shor's algorithm.

schemes are the area most vulnerable to quantum computers. Quantum cryptography is a form of secret key encryption scheme. And thirdly, Bulk. As with everything, there is always a trade-off between the ideal, and what is practical in real life. In quantum cryptography, the ideal (of unconditional security) is non-negotiable. This makes the resulting protocols bulky, unwieldy and expensive. It is probable that in time, these may be reduced, but in most situations that trade-off, the balance between cost and security, is tipped to varying extents in favour of cost. Why would you pay more to make your system more secure than it needed to be?

While quantum cryptography is probably not our best bet against the emergence of quantum computers, there will always be a market for unconditionally secure communications systems. Unfortunately this then lands us in the middle of the current ongoing wider ethical debate about encryption, back doors, and government access.

1.3.6 Quantum Key Distribution

The first seeds of using quantum physics to provide security were not dreamt up as a defence against the quantum computers, but in fact appeared the year before as a throw away comment on an interesting, if as yet far from practical, idea for banknotes which couldn't be forged [47]. It was two authors, Bennett and Brassard [48], who realised that this initial idea by Wiesner had more practical applications securing communications rather than banknotes.

There are now any number of different quantum communication schemes, each claiming higher security or better data rates than the last. But in essence they all follow the same structure. Two parties, Alice and Bob, wish to share a secret message. As we have seen in section 1.3.1 the only proven way for them to do this is via the one-time pad. The weaknesses in the one-time pad have always been with the key used, rather than the encryption itself. Quantum Key Distribution

(QKD) aims to eliminate any weaknesses in the key, ensuring both that it complies with the necessary restrictions (sufficiently random, even distribution of ones and zeroes) and that it has not been overheard by an eavesdropper (Eve). The key in itself does not convey any messages or information between Alice and Bob.

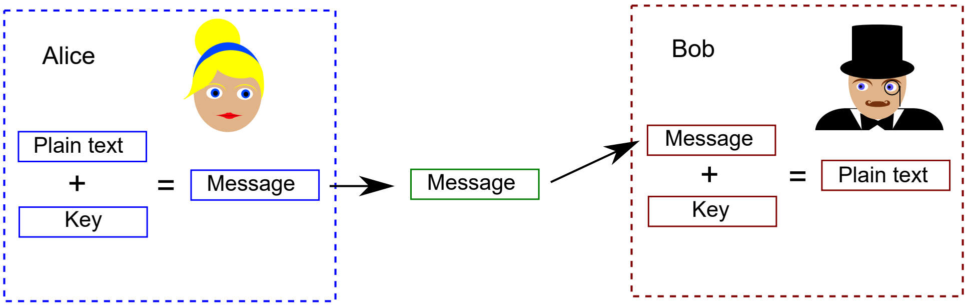


Figure 1.12: Alice adds the key to the plain text she wishes to send to Bob to create the message. Bob receives the message and adds the key to it to reveal the original plain text.

A basic overview of the protocol is as follows. Alice encrypts her plain text (the substance of which is what she wants to communicate to Bob) using the key. She then sends the resulting message to Bob, who proceeds to decrypt it using the same key, in order to obtain the original message. This is demonstrated in figure 1.12.

1.3.7 Early Protocols

The first, and most famous, QKD protocol is the BB84 protocol [48], named after its inventors, Bennett and Brassard, along with the year of its conception, 1984. This is the simplest of the protocols to understand, as it serves more as a demonstration of principle, unhampered by the multitude of practical considerations that must attend any real-world application. It is however seen by many as the gold standard, most secure, cryptosystem, and many groups try to stay as close as possible to this idea.

BB84

The BB84 protocol works as follows. Alice prepares a photon with one of four possible polarisations; horizontal, vertical, or ±45° (i.e. both diagonals), shown in figure 1.13. The rectilinear and diagonal states are non-orthogonal, which means that it is impossible obtain complete information about both of them. It is necessary to measure the rectilinear states in a rectilinear basis and the diagonal ones in the diagonal basis.

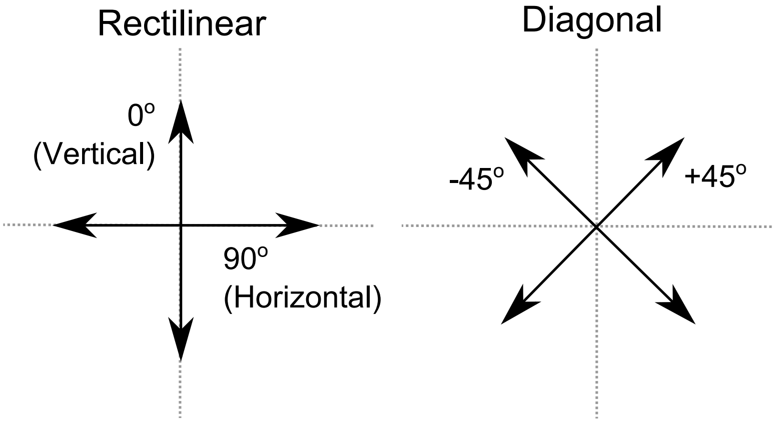


Figure 1.13: Rectilinear and diagonal photon polarisations.

If a measurement basis was chosen that was different to the state to be measured (i.e. if you tried to measure a diagonal state in a rectilinear basis, or vice versa), the measurement result would be forced into the measurement basis. Because the diagonal state is half way between the two possible outcomes of the rectilinear basis, if you tried to measure a diagonal state in the rectilinear basis, your outcomes would be horizontal 50% of the time, and vertical the other 50%. There is an equal probability that each diagonal state measured in this basis will result in either of the two rectilinear outcomes. This is demonstrated in figure 1.14. If however, you chose the matching measurement basis, you get the correct result. A horizontal state will be detected as horizontal, the vertical as vertical etc.

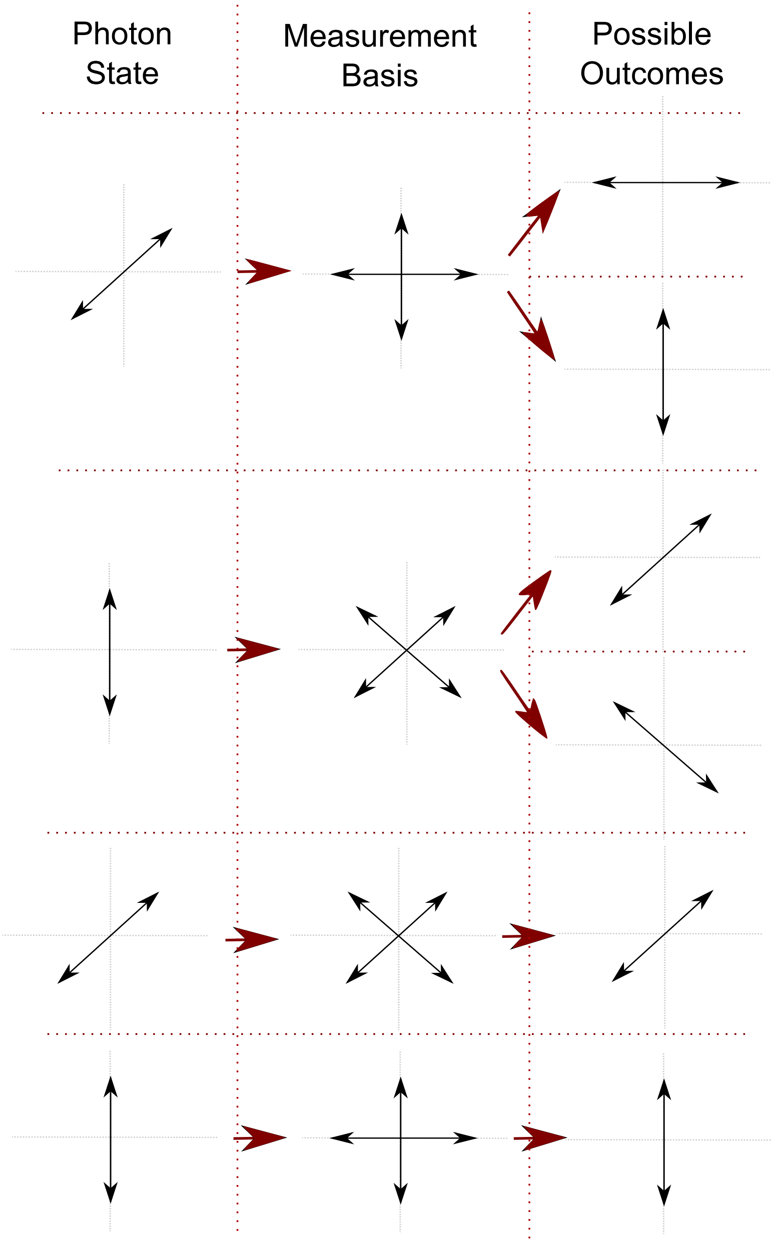


Figure 1.14: The possible measurement outcomes when choosing matching or unmatching measurement bases.

After Alice has prepared a photon in one of these four states, she sends it to Bob who chooses one of two measurement bases to measure it in. He doesn't know in which basis Alice has prepared the photon, so chooses his measurement basis randomly. Half the time he will choose wrong. At this point, he doesn't know if he has chosen the correct basis or not. All he has is his measurement outcome.

Once the quantum transmission of all the states required to make up the message has finished, Alice and Bob then communicate over a classical channel. This needs to be an authenticated, public channel, one which anyone (including an eavesdropper, Eve), can listen to, but to which no one can interfere. Alice uses this channel to tell Bob the basis of the photon she sent, but not the individual state. For example, if she sent a horizontal photon, she will say she sent something in the rectilinear basis. This gives Bob enough information to discard the photon if he measured it using the incorrect basis, but not enough information for Eve to work out which state was sent.

For an eavesdropper to intercept the message, she would need to intercept, measure, and then re-send each individual photon. However, Eve doesn't know which basis Alice prepared each state in, so she has to choose the measurement basis at random. This means that half of the time she will choose the wrong basis. When this happens, the state she sends on is different to the one originally sent by Alice. How this affects Bob, is dependent on the measurement basis he has chosen. This is best explained visually in figure 1.15.

The no-cloning theorem [49] means that Eve cannot measure the unknown quantum state in a way which does not disturb it. This is a central point in quantum cryptography as it means that not only can Eve not intercept and re-send the states without introducing errors for Bob further down the line, she also cannot create perfect copies of the states that she can then read using the correct measurement basis later, once Alice has revealed which bases her states were sent in.

<table>
<tr><td>Alice</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>Alice basis</td><td>+</td><td>×</td><td>×</td><td>+</td><td>+</td><td>+</td><td>+</td><td>×</td><td>×</td><td>×</td></tr>
<tr><td>Alice sends</td><td>|</td><td>/</td><td>\</td><td>–</td><td>|</td><td>|</td><td>|</td><td>/</td><td>\</td><td>\</td></tr>
<tr><td>Eve basis</td><td>+</td><td>×</td><td>+</td><td>×</td><td>+</td><td>×</td><td>+</td><td>+</td><td>×</td><td>+</td></tr>
<tr><td>Eve measures</td><td>|</td><td>/</td><td>–</td><td>\</td><td>|</td><td>/</td><td>|</td><td>–</td><td>\</td><td>|</td></tr>
<tr><td>Eve</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>Eve sends</td><td>|</td><td>/</td><td>–</td><td>\</td><td>|</td><td>/</td><td>|</td><td>–</td><td>\</td><td>|</td></tr>
<tr><td>Bob basis</td><td>+</td><td>+</td><td>×</td><td>×</td><td>+</td><td>+</td><td>×</td><td>+</td><td>×</td><td>+</td></tr>
<tr><td>Bob measures</td><td>|</td><td>–</td><td>/</td><td>\</td><td>|</td><td>–</td><td>\</td><td>–</td><td>\</td><td>|</td></tr>
<tr><td>Bob</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td><td>0</td></tr>
</table>

Figure 1.15: Photon and measurement bases over the course of the BB84 protocol. The grey columns show the cases where Bob's measurement basis is different to Alice's, so those bits are discarded. The red coloured bits are those which differ from the ones held by Alice.

While in theory, this is secure, there have been a number of weaknesses found, in different implementations, usually in the experimental realisation. These have mostly since been resolved, but the protocol loses a lot of its elegance. Such examples include the difficulty in achieving single photon transmission, and exploitation of the structure of single photon detectors [50–61].

EK91

In 1991 Ekert published the EK91[1][62] protocol. This uses the entanglement of photons and Bell's inequalities as the source of security.

In this scheme, the protocol starts with a photon source, which produces pairs of maximally entangled photons. One photon from each pair is sent to Alice, while the other is sent to Bob (Figure 1.16). As with BB84, each of these photons will yield either a 0 or 1 when measured. Both Alice and Bob measure their photons

[1] Sometimes EK92

simultaneously, each using an independently and randomly selected basis. Due to the nature of entanglement, on the occasions when Alice and Bob both choose the same measurement basis, they obtain the same result.

After the transmission, Alice and Bob announce publicly which basis they were using, and keep the cases where the bases matched. However, instead of throwing away the cases where the bases were mismatched, they use these to calculate the correlation coefficient. If the photons they measured were undisturbed (had not been intercepted by an eavesdropper), then this correlation coefficient would return a higher value than if they had been disturbed. This correlation coefficient is effectively a measure of the entanglement in the system. If an eavesdropper intercepts and reads one of the pair of photons, then she destroys the entanglement, and renders the system classical, meaning you observe a lower value of the correlation coefficient. This means that any eavesdropper in the system can be detected by using information which would otherwise have been thrown away.

EK91 is effectively an implementation of BB84, as they are the same up until the teleportation. These are the first two QKD schemes, and the easiest to understand. The vast majority of subsequent schemes are built from these two main building blocks.

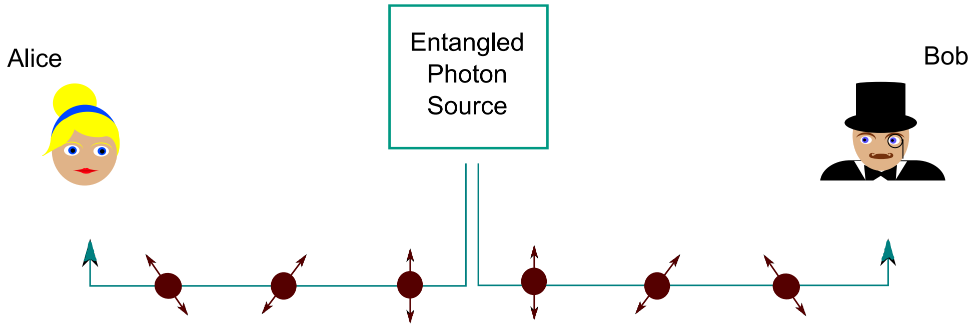


Figure 1.16: In the EK91 protocol, Alice and Bob each receive one of an entangled pair of photons.

1.3.8 Single photon protocols

Both the BB84 and EK91 protocols work with single photons. This field of single photon, or discrete variable, QKD is the larger, more trusted, and furthest developed field. Not only are there a multitude of academic protocols [63, 64], but these are spilling out into industry as well. While ID Quantique [65] is perhaps the most well-known of these, others such as Qubitekk [66] also exist. High profile QKD schemes with satellites (for example [67–69]) also help to increase excitement around the technology.

1.3.9 Coherent State QKD

Implementing single photon QKD can be challenging, as it is difficult to produce and detect single photons. In fact, most schemes use very weak coherent pulses of light with an average of less than one photon per pulse. In 1992, Bennett [70] showed that it was, in principle, possible to distribute a key using two non orthogonal states. He uses the example of the phase of two weak coherent light pulses with respect to a third, brighter, reference beam. Bob can perform a generalised measurement or POVM (positive operator valued measurement), which will either give him the correct answer or no answer. Any attempt for Eve to read the values will, much like with BB84, introduce errors.

1.3.10 Continuous variable QKD

Continuous Variable quantum key distribution (CVQKD) was first proposed by Ralph [71] in 1999. He proposed the use of multi photon, continuous variable states, rather than the single photon, discrete variable states used previously. The benefits of this are manifold. Firstly, is of course the practicality. Single photon sources and detectors are incredibly expensive and difficult to make. In contrast, for CVQKD, you can use ordinary, off the shelf, lasers and detectors. The fact that single photon sources and detectors are so difficult to make has actually led to vulnerabilities in practical experimental systems [54]. Producing single photons is difficult, so often, weak pulses of light are used which on average contain less than one photon. This means that occasionally, you will have a pulse which contains two photons. When you have a two photon pulse, then the eavesdropper could theoretically take one of those photons and keep it, measuring it in the right basis later. Alice and Bob would be unable to tell when she was doing this, thus giving her an advantage. Detecting single photons is also difficult. The most common type of detectors used for this are known as avalanche detectors. If a bright continuous wave laser beam is superimposed onto the input of the detector, then the detector stops counting photons and acts as a classical linear detector, who's output can be completely controlled by that laser. There are new protocols which take into account these attacks.

The practicality of CVQKD also has other benefits. Communications using continuous variables are well understood. This means that any attempt to integrate a CVQKD system with a classical communications system would be significantly easier. Many teams are working on building networks for quantum communications [72–75], as well as looking at how to merge QKD with existing technology [76–78]. Ralph also suggested that CVQKD systems would be more suitable to free space applications, along with having potentially much higher data rates, and fewer errors.

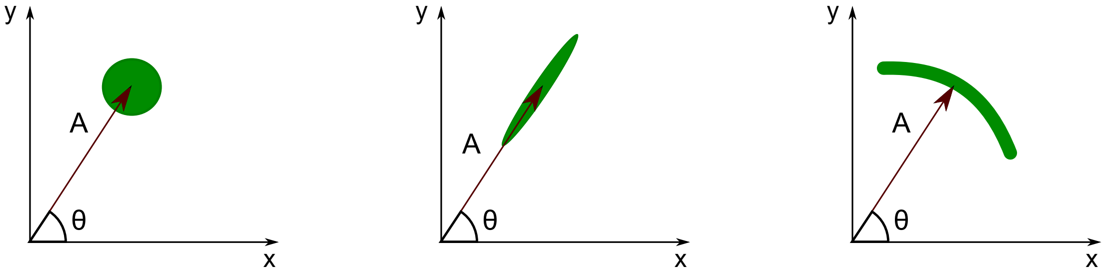


Figure 1.17: (a) The phase and amplitude of a coherent state. (b) Phase squeezed states have a very defined phase, but large fluctuations in the amplitude, and (c) Amplitude squeezed states have a well defined amplitude, but large phase fluctuations.

In CVQKD, Alice simultaneously encodes information onto the phase and amplitude quadratures of the light (Figure 1.17). Bob then performs a homodyne measurement on his received beam, measuring either the phase or amplitude of the beam at each time slot. Because phase and amplitude are conjugate variables, the amount you can know about both of them at once is limited by the Heisenberg uncertainty principle. This means that as an eavesdropper cannot know which of phase or amplitude Bob is going to measure, she has to either guess herself and risk getting it wrong, or attempt to measure both. In measuring both, she will not be able to obtain the full amount of information and will also introduce errors for Bob.

Ralph originally proposed using squeezed states of light (Figure 1.17) to increase the security of the system, but this was proved unnecessary in [79]. The Heisenberg uncertainty principle only limits the total uncertainty about the two quadratures, so it is possible to create a state with one very well-defined quadrature, but which has large fluctuations in the other. These states are known as squeezed states. It was also thought at first that CVQKD was insecure above 3dB of loss. It would be theoretically possible for Eve to replace the lossy channel between Alice and Bob with a perfect one, and introduce a beamsplitter to mimic the losses. Eve could then create a copy of the state from the perfect channel [79]. This was countered in [80] where it was shown that classical reconciliation and post-selection procedures such as advantage distillation [81], can enable key exchange above this limit when chosen well.

The CVQKD protocol which seems to have gained most standing as the basis on which others can build on and fiddle with is the one proposed by Grosshans and Grangier in [82]. In it, Alice randomly selects a phase and amplitude from a Gaussian distribution of each, and encodes them onto a laser pulse. She then sends this to Bob, who randomly chooses to measure either of the two quadratures. After many such exchanges, Bob then communicates with Alice over an authenticated public channel and informs her which quadrature he measured for each pulse. Alice then discards the irrelevant data. After this, data processing is used to create the key. This is examined in more detail in sections 2.2 and 2.3. They first have to turn the continuous data they received into a string of bits. They then compare a random subsection of their key in order to evaluate the transmission efficiency. From this it is possible for them to estimate their mutual information, and also how much information it was possible for Eve to have gained. This is important to know as it affects some of the parameters used in the following steps.

It is necessary to note here that one of the main differences between CVQKD and single photon QKD is that in single photon QKD, any eavesdropper is detected before she can gain a useful amount of information about the transmission, and the transmission is halted. In CVQKD however, it is always assumed that the eavesdropper knows some information about the transmission. The primary objective is to minimise this through any means possible down to a level where it is useless to her. A large part of this happens in the classical post-processing which happens after the quantum transmission. This makes this subsequent classical stage hugely more important in CVQKD.

Once Alice and Bob have judged the level of information that Eve might have, they then perform a reconciliation step. This is just a fancy name for error correction. In this case they use a technique called “reverse reconciliation”, where instead of Bob trying to correct his errors to match what Alice sent, Alice changes what she has to match what Bob received. After this, the data is then run through a privacy amplification scheme which reduces the amount of information Eve knows about the key.

New QKD protocols are cropping up all the time, but what makes a good QKD protocol? Secrecy is necessary. Any true QKD protocol has to be proved to be unconditionally secure. But practicality is also a major factor. Different factions focus on different measures, each trying to improve on the last. There are those who look to exchange key over the longest distances, those who compete for the highest data rate. And a few who aim for the ease of implementation. It is this last group which interests us.

1.4 Novel contributions of this thesis

This thesis is split into four main parts. The first part (chapter 1) covers the background theory on thermal and coherent light, quantum key distribution, and information theory.

Chapter 2 covers more in depth background, looking at existing protocols from which we drew inspiration for our thermal state protocol. It covers both classical and QKD protocols, as well as an in depth look at some of the classical reconciliation protocols which follow a quantum exchange. There is no novel work in this chapter.

The second part of the thesis (chapter 3) presents a new protocol based on the idea of using thermal light in a continuous variable quantum key distribution scheme. The idea of using thermal light in QKD is a novel one.

The third part of this thesis, chapter 4 describes the experimental work carried out in order to characterise the light source used in chapter 5. Again, there is no novel work in this chapter.

The final part of the thesis covers the novel work, and is formed of two chapters. The first of these is chapter 5 where we discover experimentally that using thermal light can increase security in some cases, and decrease it in others. Chapter 6 is split into two parts. Section 6.1 looks at the security of discretising the continuous variable key elements into discrete bits and the security associated with it. Section 6.2 examines the security of one commonly used classical reconciliation method, and finds situations where it is not as secure as previously thought.

This work is then summarised in chapter 7.

Chapter 2

Cryptographic Protocols

This chapter explores some of the more detailed background theory, focusing on specific quantum and classical cryptographic protocols, which influence the protocol we develop in chapter 3. We then move on to examine the classical reconciliation steps common to such protocols.

2.1 Central Broadcast schemes

In this section, we start by examining a new set of continuous variable quantum key distribution protocols which do not require information to be encoded on both quadratures of a laser beam. These new protocols only use a single quadrature, meaning physical implementations could be much easier.

This new experimental simplicity opens the avenue to different forms of key distribution, for example central broadcast schemes, rather than the traditional point to point.

2.1.1 Uni-Dimensional CVQKD

After Grosshans *et al.* published their paper demonstrating a CVQKD system [82], subsequent papers emerged [83, 84] showing that Bob did not need to randomly switch between the two quadratures, but could in fact gain a higher data rate by simultaneously measuring both. This also eliminated the tricky practical step of needing to keep the phase of the local oscillator[1] precisely controlled during the switching. They find that due to the joint Heisenberg relation between Alice and Bob, and Eve and Bob, there is a limit to what Alice and Eve can both know about Bobs measurement results. This gives them a good estimate of Eves knowledge of Bobs measurements, and also allows them to double the data rate by using the information encoded onto both quadratures rather than binning one.

This has been taken even further in [85, 86], who proposed that in fact, it is only actually necessary to encode information onto one of the quadratures. The protocols they proposed are very similar[2], and run as follows. Alice displaces one of her quadratures (let's say amplitude for convenience), according to a random Gaussian variable. She then sends this to Bob. Bob measures the amplitude most of the time, but just occasionally, will randomly measure the phase instead. If an eavesdropper were to be measuring the amplitude of the states, she would destroy the phase. This monitoring of the phase by Bob, ensures that if this does happen, then they will know. These were experimentally realised in [87].

2.1.2 Classical central broadcast schemes

One of the advantages of CVQKD over single photon QKD is its much higher tolerance to noise. As you don't need to preserve the states of a large number of single photons, it can deal with much higher levels of environmental or channel noise. CVQKD is also currently being tested in the microwave regime by a number of groups, meaning it may not be limited to optical fibre or line of sight transmission. This makes it more suitable for use in a number of existing communications set-ups.

[1] The local oscillator is a reference beam, taken from Alices beam before she adds her modulation, and sent on to Bob. Bob then uses this to measure the phase of the signal beam.

[2] [85] is by far the better written article, I would advise not reading [86] if you want your brain to stay intact.

Apart from the obvious example of optical fibre, most of these existing communications schemes have a tendency towards a central broadcast model (figure 2.1). This is where a transmitter, such as a Wi-Fi router, a mobile phone or radio mast, or even a satellite, transmits a signal to cover a wide area. Part of this signal is then picked up by a small receiver, such as a laptop, phone, radio or GPS device[1]. While, at first glance, it seems that it would be foolish to attempt to distribute a secret key over such a central broadcast scheme, it turns out that this could actually be possible.

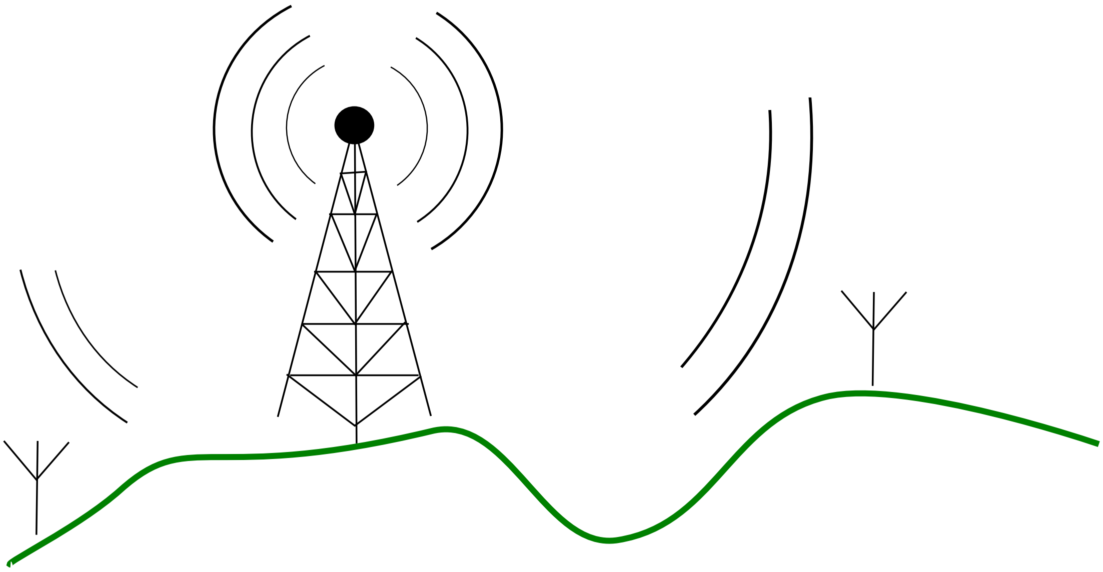


Figure 2.1: Central broadcast schemes consist of a transmitter which transmits a signal over a wide area, which is then picked up by receivers.

In fact, in 1993 Maurer found that as long as the two legitimate parties, Alice and Bob, had access to an authenticated public channel[2], then they could use a

[1] Most of these technologies use frequencies in, or close to, the microwave regime. The discussion about the suitability of microwaves for CVQKD is ongoing and not the subject of this work. The debate on this subject is sufficiently lively that publications are scarce, its still in the pistols at dawn during conferences stage. Enter at your peril.

[2] One for which Eve can hear all discussion, but cannot interfere in any way. (For example, she cannot pretend to be one of the legitimate parties or alter the messages whilst in transit).

central broadcast system to distribute a secret key classically [81].

This story begins with the work of Wyner [88], who showed that Alice can send information to Bob in perfect secrecy over a discrete memoryless channel[1], without the need of the additional authenticated channel, providing some restrictions were placed on the eavesdropper, Eve. These were that Eve could only access the channel via a wiretap which occurred at the end of Bob's channel. This wiretap fed into a second, noisy channel, meaning that Eve had not only the noise on the main channel to contend with, but that on her own as well. So basically, Alice and Bob can exchange secret information over a discrete memoryless channel providing that Eve has more noise than Alice and Bob.

This is most easily explained with an example of the special case where Alice and Bob have no noise, rather then an arbitrary amount, less than Eve. First, Alice produces a binary variable, S, with $\Pr\{S = 0\} = \Pr\{S = 1\} = \frac{1}{2}$, equal probability of being 0 or 1. She builds these into a string S^K, of length K. This string is passed through an encoder, which works as follows. Each bit of S entering the encoder becomes a binary string of length N. If the bit of S is 0, then the encoder output X^N is a randomly chosen vector with an even parity (an even number of ones). If S is 1, then X^N is a randomly chosen vector with an odd parity. Alice sends the vector X^N to Bob through a noiseless channel. Bob then decodes X^N into $\hat{S}^K$. As the channel they use is noiseless, Bob's decoder works perfectly, and $\hat{S}^K = S^K$. The probability of error ($P_e = \frac{1}{K}\sum_{k=1}^{K} \Pr\{S^k \neq \hat{S}^k\}$) is 0.

Eve has a wire tap at the end of the channel between Alice and Bob. This wire tap feeds directly into a binary symmetric channel which has a crossover probability, p_0, of $0 \leq p_0 \leq \frac{1}{2}$, and output Z^N. If z is a vector of even parity,

[1] One in which the inputs and outputs are discrete rather than continuous, and the outputs do not depend on any previous inputs.

then,

$$\begin{aligned}
\Pr\{S=0|Z^N=z\} &= \Pr\{\mathrm{Eve's\ channel\ has\ even\ number\ of\ errors}\} && (2.1)\\
&= \sum_{j=0,\mathrm{even}}^{N} \binom{N}{j} p_0^j (1-p_0)^{N-j} && (2.2)\\
&= \frac{1}{2}+\frac{1}{2}(1-2p_0)^N. && (2.3)
\end{aligned}$$

Similarly for the odd cases,

$$\begin{aligned}
\Pr\{S=0|Z^N=z\} &= \Pr\{\mathrm{Eve's\ channel\ has\ odd\ number\ of\ errors}\} && (2.4)\\
&= \frac{1}{2}-\frac{1}{2}(1-2p_0)^N. && (2.5)
\end{aligned}$$

So for all $z \in 0,1^N$, $H(S|Z^N = z) = h[\frac{1}{2} - \frac{1}{2}(1-2p_0)^N]$, where $h(\lambda) = -\lambda \log \lambda - (1-\lambda)\log(1-\lambda)$, $0 \leq \lambda \leq 1$. This means that the amount of confusion Eve has, $\Delta = H(S|Z^N) = h[\frac{1}{2} - \frac{1}{2}(1-2p_0)^N] \to 1 = H(S)$ as $N \to \infty$. So as the length of the encrypted string approaches infinity, Eve's confusion approaches the unconditional source entropy, and the communication can be held in perfect secrecy.

Csiszár and Körner [89] then generalised Wyner's results for a discrete memoryless broadcast channel, (as opposed to the wire channel) and found, again, that Alice and Bob could share secret information, provided that Eve's channel was noisier than Alice and Bob's. For example if Eve were to have noisier receiving equipment, or more environmental noise.

Their broadcast system was modelled as follows. Alice sends a common broadcast message, T to both Bob and Eve. She also wants to send a private message, S, to Bob, so that it remains secret from Eve. Alice uses a deterministic block encoder to map $f : S \times T \to X^n$. The message X^n is broadcast to both Bob and Eve. The system uses two decoders, comprising of a pair of mappings. Bob's, which reveals both the private and the common message, $\varphi : Y^n \to S \times T$, and Eve's, which only reveals the common message $\psi : Z^n \to T$. The encoder-decoder (f, φ, ψ) allows an (n, ε)-transmission if and only if for every $s \in S, t \in T$, Bob's decoder φ gives the correct (s, t) and Eve's decoder ψ gives the correct t with a probability $\geq 1 - \varepsilon$.

It was Maurer's introduction of an authenticated public channel for Alice and Bob to use after the initial transmission, which enabled Alice and Bob to share secret information over a central broadcast channel, even if Eve had less noise than Alice and Bob. This is an interesting result, and at first glance can seem counter-intuitive.

As with everything in cryptography, the assumptions made are key. Here, the assumptions we need to look out for are those involving noise. Especially the noise at the eavesdropper.

Maurer started with that best of all places to start, Shannon. There are three main points which Shannon argues are needed for secret communications [17]. These are:

1. For a cypher to be secure, the cyphertext must give no information about the plain text.
2. For this to happen, the key must be at least as long as the plain text.
3. It is assumed that Eve has access to all the information available to Alice and Bob, except for the secret key.

It is this last assumption, (point 3) which Maurer takes issue with. In theory, yes, an Eavesdropper limited only by the laws of physics can gain this much information. However, in real life, this is not true. In any real-world system, noise is always present.

In the classical world, noise is everywhere. There is noise in preparation, several types of noise can interfere during the transmission, and then there is always noise in the receiver as well. However, in this new quantum world, where we consider really, really powerful eavesdroppers, these limitations are no longer always appropriate. It is assumed than an Eavesdropper has access to a vast array of incredibly powerful as yet unimagined future technology. She is limited only by the laws of physics. Handily though, the laws of physics can be pretty strict. Take for example, shot noise. This will affect Eve's detector whether she wants it to or not, and, importantly, it is independent of the shot noise in Bob's detector.

This leads us nicely onto Maurer's second assumption, which is that Alice, Bob and Eve all have independent noise. Again, on the surface, this is very easy to argue with. However, on closer inspection, while yes, there are some sources of noise that will affect all parties equally, in reality, this is of course not going to be the case for every source of noise.

Maurer describes an example of a system, where instead of Alice broadcasting to Bob (and Eve), all three parties are receiving from some fourth party broadcaster (figure 2.2). Details about the security when the broadcast station is trusted or untrusted are discussed later.

He uses the example of a satellite, transmitting a sequence of random bits using binary antipodal signalling. (Simply ones and zeroes using the extremes of the transmission state). The three channels are each subject to independent additive white Gaussian noise (AWGN). The noise on each channel is statistically independent to the noise on each of the other channels.

While the bits are sent in binary, they are received with analogue receivers. This means, that up to a certain level of noise, Alice and Bob can tell whether they have received a bit well or not. This allows them to set a confidence value for each bit, and evaluate their overall channel noise.

After transmission, Alice and Bob then publicly discuss which bits they both received well (without revealing the value), and bin the rest. During this post-selection, Alice and Bob gain the advantage because they have selected the bits that benefit them. Eve hasn't necessarily received these bits with little noise as well. There should at least be some bits in that selection that Alice and Bob

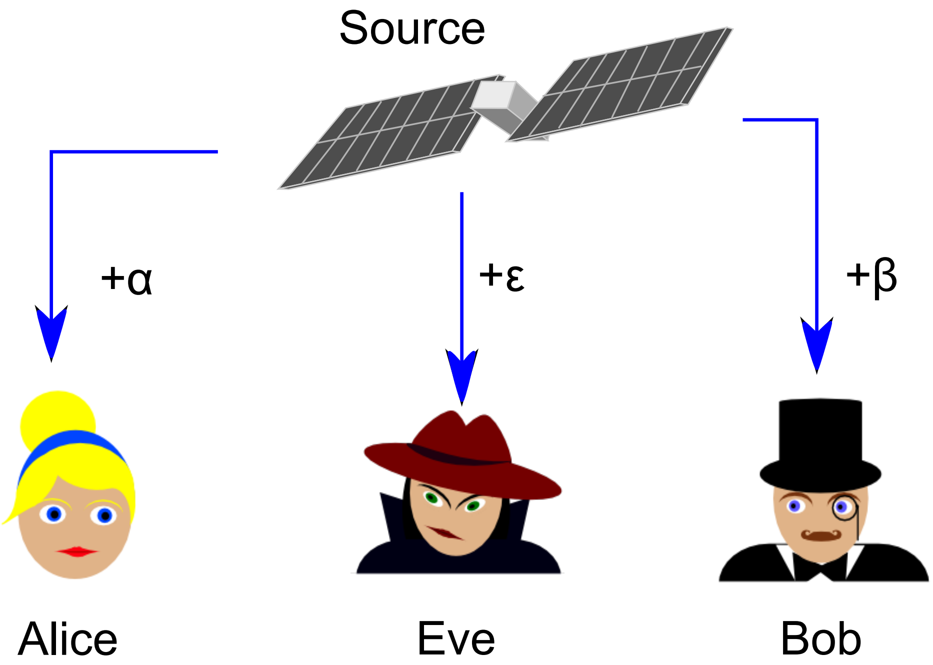


Figure 2.2: A central broadcast scheme where Alice does not control the source. Each participant is subject to their own, independent noise α, β, or ϵ.

know better than Eve.

Due to the fact that the three of them have independent noise, Eve cannot deduce the values of all the chosen bits by knowing only their position. This is because she will have high levels of noise on some of those bits.

Alice and Bob can then use further advantage distillation techniques (explored in section 2.3) to further reduce Eve's knowledge of the key. By using this simple post selection technique on this entirely classical central broadcast system, Alice and Bob can share some secret information. All they need is for the eavesdropper to have some noise, even if that noise level is less than that of Alice and Bob.

There are a few points here that are worth clearing up. In this scenario, Alice does not control the source. This is not a prepare and measure scheme. The source however, does need to be trusted. This is because the secrecy of this scheme relies on there being bits which Alice and Bob know to a high probability, but which Eve has received with too much noise. It is important that Eve does not know the value of these bits, as she already knows their location. Were Eve to control the source, she would know the values of all the bits - in effect she would have no noise, so the scheme would collapse.

A security analysis of using a very similar central broadcast scheme, adapted for CVQKD is currently being undertaken by Ghesquière. The first part of this can be found in [90]. The main difference between this and the original scheme is the change from a discrete channel to a continuous one. This causes problems with the post-selection as it becomes impossible for Alice and Bob to tell how much error they received on each bit, so are unable to choose only the low error bits to form the key from. Instead, they have to rely heavily on methods of discretisation (see section 6.1) and error correction as well as the original advantage distillation codes, then additional privacy amplification (section 2.3). The main point of the scheme however, remains the same. That there are some bits that Alice and Bob have received correctly, that Eve hasn't, and it is a matter of finding these and then using them to your advantage to distil and then amplify a secret key from the swamp of original data. It is clear, however, that this is a slow and wasteful way to produce a key, although it is relatively simple and very cheap and easy to implement. It is also still unclear how well it will work in the high noise regime.

2.2 Discretisation

Continuous variable quantum key distribution (CVQKD) schemes all involve the transmission of continuous variables. At some point, these continuous variables need to be discretised into a binary alphabet, in order to be physically useful to the participants. For example, most common encryption schemes (including the one-time pad) require binary inputs for the message and key.

Most people translate their key into binary directly after the measurement process in the key distribution. This makes it significantly easier to perform any classical reconciliation and privacy amplification which may be necessary. While it is possible to perform these tasks on a continuous key, it is generally deemed reasonably impractical and has not been widely adopted.

There are currently two main ways of discretising the measured values. The first of these is to very simply divide the range of measured values through the middle, with values falling to the left of this line translating to zero, and values to the right, one. The second method, described by Van Assche *et al.* in [91] is somewhat more complicated. They observe that when using the first method, each transmitted point becomes a single bit of key. They instead propose a scenario where each transmitted point can produce several key bits, thus significantly enhancing the key rate without altering the physical transmission in any way. They also combine this discretisation with an error correction protocol, to try and reduce the overall amount of information leaked to an eavesdropper.

2.2.1 Sliced Error Correction

Sliced Error Correction (SEC) starts after the quantum transmission, where Alice and Bob each possess a Gaussian distribution of continuous variables (X and X' respectively). Firstly, Alice and Bob compare a small subset of their data. This enables them to estimate the signal to noise ratio (SNR) of the channel, along with the joint probability density $f_{X,X'}(x,x')$. They then choose (based on a balance between efficiency, computing power, and the SNR) a number of "slices", m.

Alice divides her Gaussian distribution of X into 2^m intervals. The placing of these intervals is determined by an interval labelling function $T(X)$, which is chosen to maximise $I(T(X);X')$, the mutual information between the labelled points and Bob's raw key bits. For example, placing the intervals at equal widths reduces the effect of Gaussian noise, while placing the intervals so that they're evenly distributed throughout the data (the same number of data points fall into each interval), maximises the entropy.

Alice then creates m "slice functions" by assigning bit values to each interval. The least significant bit value is assigned to the first slice function, S_1, while the most significant bit is assigned to the last, S_m. This is best explained pictorially, in figure 2.3.

Bob then uses a "slice estimator function", $\tilde{S}(X')$, to try and recreate Alice's first slice, S_1. For the first slice, the only knowledge he has to help him are his values of X'. With his knowledge of m and $f_{X,X'}(x,x')$, he can also recreate $T(X)$, and assign bit values to his raw key. This gives him $\tilde{S}_1(X')$. Alice and Bob then perform an error correction protocol, for example cascade (See section 2.3.3), so that Bob can correct $\tilde{S}_1(X')$ to match $S_1(X)$.

The second round proceeds exactly as the first, but this time, Bob can use his corrected values from the first round $S_1(X)$, to help him guess the values for the second, $\tilde{S}_2(X')$. For example, if he had measured a raw key element to be in interval 4, then if say $m = 5$, he would assign that element a value of

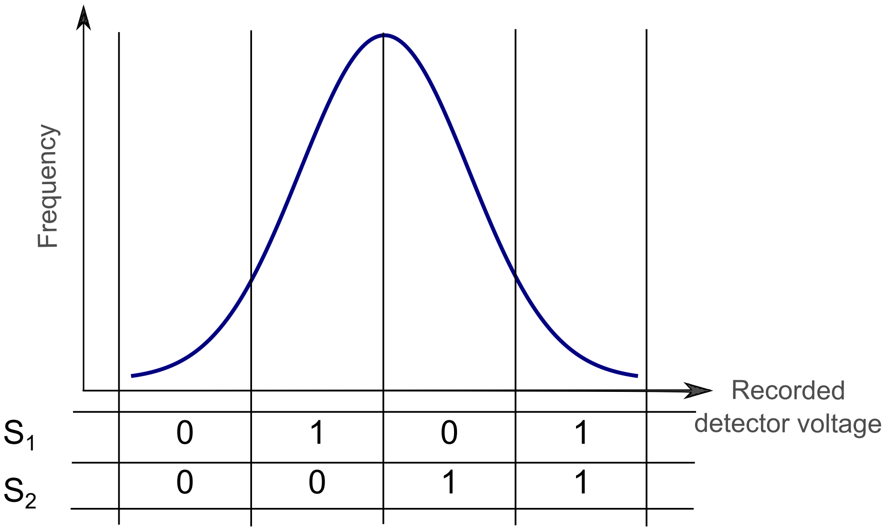


Figure 2.3: Bit assignment in sliced error correction

00100. If this key element had been subject to noise, and had started life in a position such that Alice had put it into the 3rd interval, then she would have assigned it a value of 11000. Now, in the previous round, Bob corrected that first bit to a one. He knows that he has some noise on that element. If the SNR was small, he would know that it likely only shifted a small number of bins, and can therefore guess the value of the subsequent bits more accurately. It is unclear if this technique does in fact provide an advantage over performing the error correction separately, especially in the presence of high levels of noise. Indeed in [92], the authors provide an example, using an SNR of 15, and 5 slices. They state that in this case, the first two slices are sufficiently uncorrelated for the error correction algorithm to fail, indeed "it is enough for Alice to entirely reveal $S_1(X)$ and $S_2(X)$ for the whole string." It seems unlikely that this is an optimal strategy, as they are effectively revealing to the eavesdropper 2/5 of the key. There is a limit to the effectiveness of Privacy Amplification protocols[1].

[1] See section 2.3.5.

2.2.2 Problems in Discretisation

Is this really the best way of performing discretisation of the raw key elements? We examine a number of factors which may influence either the key rate or the amount of information leaked to an eavesdropper, and try to find a simple and coherent strategy for discretising the key.

This can be broken down into a number of different categories. The first question is, of course, “How many intervals, or bins, should the data be split into?”. The second, obvious, question is “Where should those bins be positioned?”, and the third, less obvious question is “Is there an optimal way of labelling those bins?”.

Throughout, we must bear in mind the fact that what we are trying to do here is not only to maximise the mutual information between the two legitimate parties, Alice and Bob, but also to minimise the amount of information known by an eavesdropper. This is perhaps a little counter-intuitive. After all, during the discretisation process no information is exchanged. How could the choice of discretisation process affect the eavesdropper?

The answer is very simple. In exactly the same way as the choice of discretisation method can affect the mutual information shared between Alice and Bob, it can also affect the mutual information between Bob and Eve.

2.2.3 Measurement in CVQKD

We start by looking at the process of measurement itself, as it is essential to fully understand what is happening, so that any protocols envisioned make sense in a practical environment.

2.2.4 Transmission

In CVQKD protocols, during the quantum transmission, Alice sends a signal made up of a series of pulses to Bob. Each of these pulses is a Gaussian state which falls somewhere on a Gaussian distribution of values. As each pulse is sent to Bob, it may change slightly as a result of preparation noise, noise from the channel, or detector noise at Bob's end. Bob then measures this signal.

The possible values Alice sends are continuous, or indiscrete. At least to the resolution of her transmitter. However, by the time the pulses have had noise from the above mentioned sources added, they will be continuous. Bob receives a series of pulses from Alice, which make up a continuous distribution.

2.2.5 Detection

Most detectors work on the basic principle of photons hitting a photodiode and producing electrons (see section 4.2.1 for details). The more photons hit the detector, the more electrons are produced. When coupled to a load resistance in an oscilloscope, or other data acquisition device, this photodiode current becomes a voltage, proportional to the intensity of light hitting the detector. It is in the second, data acquisition equipment, that the first discretisation occurs.

2.2.6 Initial Discretisation

In this digital age, at some point in the detection process there will exist an analogue-to-digital converter (ADC). This quantises the analogue signal of the detector itself, turning the continuous signal sent by Alice into a number of discrete bits which can be understood by a computer. All the noise associated with this process in CV QKD is generally included as part of the detector noise.

While there are many different types of ADC, which all work slightly differently, the main principles of operation are as follows: Firstly, the analogue signal is sampled by a track-and-hold circuit. This takes a sample from the signal, and maintains it at that level for sufficient time for the next step to complete. During this next step, the input voltage is compared to a series of reference voltages, and the closest match is chosen. This is then encoded digitally and passed onwards in the detection process.

This results in the most precise possible reading.

2.2.7 Further Discretisation

While these values are now in binary, they are still not particularly useful for CVQDK, due to the precision of the reading.

In a high noise environment, very precise readings are not necessarily useful for Alice and Bob. In fact, it is harder for Alice and Bob to reconcile a key if their values are particularly precise. The more precise the values, the more digits used to describe them, the more likely there is to be a difference between the two, due to the noise. Alice and Bob therefore have to "round" their values to a lower precision in order for them to agree a key without giving away too much information to an Eavesdropper.

It is this further level of discretisation which we concern ourselves with.

2.2.8 Classical Inspiration

The problem of trying to descretise a Gaussian of continuous variables in the presence of noise is not inherently a quantum one. The measurement of the quantum states sent across the transmission channel happens at the detector. After detection, or measurement, these states can be treated classically. The "quantumness" of the states cannot persist past measurement.

This means that all we are trying to do in this step is to assign bit values to a Gaussian distribution of continuous variables, in such a way as to minimise the uncertainty between Alice and Bob, while ensuring that as little information as possible is revealed to an eavesdropper.

Classical communications theory is a very well established and well understood field. It makes a great deal of sense to borrow and learn from them rather than trying to reinvent the wheel.

2.2.9 Shannon-Hartley Theorem

This is more commonly known amongst the QKD community as the Shannon channel capacity. In the original formula, designed for normal communications systems, the channel capacity, C, is the maximum bit rate of the channel, and is given by

$$C = B \log_2 \left(1 + \frac{S}{N}\right), \tag{2.6}$$

where B is the bandwidth of the channel, and $\frac{S}{N}$ the signal to noise ratio, or SNR.

The form of this equation most commonly seen in the QKD literature tends to be similar to

$$I = \frac{1}{2} \log_2 \left(1 + \frac{S}{N}\right), \tag{2.7}$$

where I is the information rate in bits per symbol (or bits per key element).

It is important to note that these are for additive white Gaussian noise (AWGN) channels.

2.2.10 Hartley's Law

This is actually a precursor to the Shannon-Hartley theorem, which very simply tells us the maximum number of bits, M, which can be reliably sent through a channel and still be distinguished at the other end.

$$M = 1 + \frac{A}{\Delta V}, \tag{2.8}$$

where A is the range of signal amplitudes, and ΔV the precision of the detector. Or, in a slightly wider definition, A is the range of possible received values, and ΔV the spread, or uncertainty associated with them. The principle of this is shown in figure 2.4.

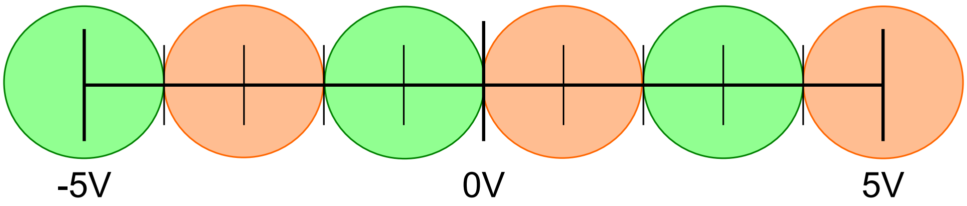


Figure 2.4: An example of Hartely's law, where $A = 10$ and $\Delta V = 2$.

This is very intuitive for a classical channel, where you are aiming principally to be able to easily distinguish bits, and secondarily to maximise the number of bits. However, these are not the same aims as those needed for a quantum cryptographic channel. Here, the principal aim is to prevent eavesdropping, and having easily distinguishable bits will not help this. While Hartley's law may help to maximise the mutual information between Alice and Bob, it is likely to do so at the cost of their secrecy.

2.2.11 Lloyd's Algorithm

In 1982, Lloyd published a description of an algorithm [93] that divides a Gaussian distribution of signal values into intervals in such a way as to minimise the amount of noise[1]. This is very similar to what we wish to achieve - dividing Bob's Gaussian distribution of received signal amplitudes into a number of intervals. While also minimising the effect of the channel noise. Unfortunately, as with a lot of these things, we *also* want to do this in a way which minimises the usefulness of any information the eavesdropper has gained during the quantum transmission, or may gain in the subsequent reconciliation stages. The details of this are given in the methods section (6.1.2).

The idea of using Lloyd's algorithm has been suggested in the context of CVQKD before by [94], building on work done in [95, 96], which relate mainly to data compression. A review of quantisation methods up until 1998 can be found in [97]. All of these methods are based on a two party situation, trying to get one data set as close as possible to a second. None of them consider a situation where secrecy is important.

2.3 Reconciliation

In all real physical systems, you will have noise and errors. This is an unavoidable fact of life. To deal with these, the quantum exchange is usually followed by a classical reconciliation stage, where some information is exchanged between Alice and Bob classically in order to try and remove these errors. This is assumed to be done over an authenticated public channel - one where the Eavesdropper has complete access to everything that is shared, but cannot interfere. For example, she cannot pretend to be one of the legitimate parties.

In theory, quantum cryptographic protocols can detect the presence of an eavesdropper, they are frequently limited to detecting her presence only if she takes enough information to be useful to her.

This means that after the quantum transmission, there will be some errors between Alice and Bob, and Eve could have knowledge of some portion of that

[1] For the avoidance of confusion: Lloyd uses the word "quanta" to mean what we have described here as "intervals" or "bins".

transmission.

The classical communication stage needs to both correct the errors between the two legitimate parties, whilst also ensuring that the eavesdropper cannot gain sufficient information about the final key. There are three main steps to this classical stage:

1. Advantage Distillation - This is used to create an information advantage over the Eavesdropper. Increasing the mutual information between Alice and Bob, while reducing it for Eve.
2. Information Reconciliation - Correcting any errors between Alice and Bob. Care has to be taken not to give away additional information to Eve during this process.
3. Privacy Amplification - Similar to advantage distillation, except it generally happens after the reconciliation stage.

As with the quantum part of the procedure, it is assumed that the eavesdropper knows the ins and outs of the mechanisms of all these protocols.

2.3.1 Advantage distillation

The term "advantage distillation" is usually used to describe the protocols envisioned by Maurer in a series of papers and conference articles containing the phrase "Secret key agreement by public discussion" [81]. He describes how Alice and Bob can use the authenticity of the classical channel to gain an information advantage over Eve.

Alice and Bob divide their strings of key elements they've received through the quantum transmission into blocks of bits. Alice XORs each of her bits in the first block with a 0 or 1 produced by a random number generator[1]. She sends

[1] In the original suggestion, this was left open for any error correcting code to be used, but these repetition codes seem to be the norm.

the resulting block of numbers to Bob. Bob XORs this block with his own corresponding block of key elements. If there was no error, then he would receive identical copies of the same digit. This would have the same value as the digit produced by Alice's random number generator. This becomes their key element (Figure 2.5).

If there was an error in the block, then that digit would be different to the rest after Bob's XORing process, and Alice and Bob can then choose whether to discard the block, or to proceed with majority voting. This choice, along with the choice of initial block size, is chosen based on the error rate between Alice and Bob, as well as the estimated error rate of Eve[1]. The smaller the error rate, the larger the size of the blocks.

This is then repeated for each block.

The best strategy for the eavesdropper here is to proceed in exactly the same way as Bob, but not to commit absolutely to any decisions. This will be discussed further in section 6.2.1, but essentially it is wise for the eavesdropper not to throw out any information at all.

In a more formal wording, Alice sends Bob a codeword V^N, of length N, over an authenticated public channel. Bob receives V^N with bit error probability $Er_B = \epsilon_A + \epsilon_B - 2\epsilon_A\epsilon_B$. Eve, who also has access to this channel receives V^N with bit error probability $Er_E = \epsilon_A + \epsilon_E - 2\epsilon_A\epsilon_E$. Where ϵ_A, ϵ_B and ϵ_E are the bit error probabilities of Alice, Bob and Eve's measurement of the key respectively.

$Er_E < Er_B$ unless $\epsilon_E \geq \epsilon_B$. Eve's bit error rate from this protocol will be less than Bob's unless she has more errors in her knowledge of the original key distribution. Maurer shows that this doesn't hold if Bob chooses to only accept a word if the Hamming distance between it and the codeword is less than half

[1] Again, in the original proposal, Maurer suggests Bob accepts the block if it is obvious what the outcome is, whereas others (including Liu [98]) only let Bob accept the block if there are no errors. This seems to be a matter of personal choice.

the codes minimum distance. He post selects to only include the measurements he has a high degree of confidence in.

Take the example of a code of length N, with two codewords, $[0, 0, \dots 0]$ and $[1, 1, \dots 1]$. For $j = 1, 2, \dots N$, Alice generates a random bit R_j. She sends $V_j^N = [R_j, R_j, \dots R_j]$ to Bob. Bob accepts this only if it is equal to one of the codewords.

We now define $\delta_A = 1 - \epsilon_A$, and similarly for B and E. If Bob has no errors in his codeword, $P_{Correct} = (\delta_A \delta_B + \epsilon_A \epsilon_B)^N$. If Bob has all errors, $P_{Error} = (1 - \delta_A \delta_B - \epsilon_A \epsilon_B)^N$. So the probability of Bob accepting a codeword is $P_{Accept} = P_{Correct} + P_{Error}$. He will think he has received the correct code word if he either has no errors in the bits that make up the codeword, or, if every bit is in error. The channel is effectively a binary symmetric channel with bit error probability $\beta = \frac{P_{Error}}{P_{Accept}}$.

α_{rs} where $r, s \in \{0, 1\}$ is the probability that a 0 sent by Alice is received by Bob as r, and Eve as s.

$$\alpha_{00} = \delta_A \delta_B \delta_E + \epsilon_A \epsilon_B \epsilon_E \tag{2.9}$$

$$\alpha_{01} = \delta_A \delta_B \epsilon_E + \epsilon_A \epsilon_B \delta_E \tag{2.10}$$

$$\alpha_{10} = \delta_A \epsilon_B \delta_E + \epsilon_A \delta_B \epsilon_E \tag{2.11}$$

$$\alpha_{11} = \delta_A \epsilon_B \epsilon_E + \epsilon_A \delta_B \delta_E . \tag{2.12}$$

P_ω where $0 \leq \omega \leq N$, is the probability that the codeword sent by Alice is accepted by Bob, with Eve receiving a word of Hamming weight ω. This can be written $P_\omega = \alpha_{00}^{N-\omega} \alpha_{01}^{\omega} + \alpha_{10}^{N-\omega} \alpha_{11}^{\omega}$. Eve's average error when guessing the bit sent by Alice is then $\gamma = \frac{1}{P_{Accept} \sum_{\omega=\frac{N}{2}}^{N} \binom{N}{\omega} P_\omega}$. It is always possible to choose N so that $\gamma > \beta$, Eve has more error than Bob.

This technique works as Alice and Bob can discard any information they are not too happy about. They are effectively post-selecting the cases where they have more information than Eve. It is Alice and Bob's choice which bits they

Alice			Bob			
Key block	Random No.	XOR	Received	Key block	XOR	Key element
0010	1	1101	1101	0010	1111	1
0100	0	0100	0100	0100	0000	0
1110	1	0001	0001	1010	1011	-
			Eve			
			Received	Key block	XOR	Key element
			1101	0111	1010	?
			0100	0110	0010	?
			0001	1110	1111	-

Figure 2.5: Advantage distillation enables Alice and Bob to discard some of their errors, whilst increasing the impact of Eve's.

keep.

The point of this is to try and reduce Eve's knowledge of the key, and eliminate some of the errors between Alice and Bob before entering into the reconciliation phase as reconciliation does leek some information to the eavesdropper, and can struggle with high error rates between Alice and Bob. This stage is often considered unnecessary for particular individual QKD schemes.

Bit pair iteration protocols

In the case where Eve's error rate is very small, an incredibly large block size is required, and advantage distillation protocol becomes hugely inefficient. In order to avoid this, a similar protocol was developed [99] which worked using successive iterations of the protocol using a block size of two bits.

After the first round, where the whole pair was discarded if the parity didn't match, one bit from each remaining pair was also discarded to preserve the secrecy. The bits which survived this cull were then formed into pairs and the process repeated.

2.3.2 Information reconciliation

Although the term “reconciliation” is often used to refer to the entire classical procedure, it is more correctly used to mean the error correction proceedure. There are two main error correction protocols used in QKD, the first, cascade [100], was for a long time the *de facto* technique, but due to the fact that it is particularly inefficient, it has since been replaced by low density parity check (LDPC) codes [101, 102].

2.3.3 Cascade

As with most of these classical protocols, cascade begins with Alice and Bob publicly exchanging a subset of their data in order to estimate the error rate. Because Eve now knows these values, this subset is deleted from the final key.

Cascade should not be used if you enter with an error rate $> 15\%$.

Alice and Bob start off by choosing a block size based on their error rate. This is governed by the equation

$$E = kp - \frac{1-(1-2p)^k}{2} \tag{2.13}$$

Where p is the bit error rate, the probability of an error in the block, E, the number of errors expected in a block after the first pass is less than or equal to 0.35, and k is the block size [100]. If you have too many errors in the block after the first pass, the protocol is insecure as too much information is exchanged between Alice and Bob (and thus known to Eve) in finding and correcting the errors.

The reason you can’t enter cascade with bit error probability $p \geq 0.15$ is that for this value of p, the block size k, is 5. If you had a block size smaller than 5,

	Alice								Bob							
Block	0	1	1	0	0	1	1	0	0	1	1	0	1	1	1	0
Parity				(0)								(1)				
$\frac{1}{2}$ Block	0	1	1	0					0	1	1	0				
Parity				(0)								(0)				
$\frac{1}{4}$ Block					0	1							1	1		
Parity				(1)								(0)				
$\frac{1}{8}$ Block					0								1			
Parity				(0)								(1)				

Figure 2.6: Parity comparisons in cascade. The parity of each block is compared, before splitting the block to narrow down the mis-matched parity and locate the error.

then it becomes too easy for Eve to simply guess the parity.

Alice and Bob divide their strings into blocks of the required block size. Alice then sends the parity (sum modulo 2) of each of her blocks to Bob. Bob looks to see if these parities sent by Alice match the parities of his corresponding blocks. If they do match, then there must be an even number (or 0) errors in that block. Nothing further is done with those blocks. If, however, the parities do not match, then they know that there must be an odd number of errors. Alice sends Bob the parity of the first half of that block. If the parities for the first half match, then an odd number of errors must exist in the second half. Alice sends the parity of the first half of that half, etc. until an error is found (figure 2.6).

In earlier versions of the protocol [103], this error was deleted. However, in this version, the error is corrected as it enables a much higher reconciliation rate, at a cost of an increase to the amount of information leaked to an eavesdropper. The ability to correct errors is, after all, the point of a reconciliation protocol. Increasing the amount of leaked information, purely passes the burden of dealing with this leaked information to other protocols such as privacy amplification. Unfortunately, these too have their limits, and it's crucial to strike the right balance.

As there are no further error correction steps, it is important to correct as many errors as possible here.

At this stage all of Bob's blocks have an even number of errors (as the parity check can only find odd errors). Alice and Bob choose a random function that re-arranges the order of their bits (colloquially known as a "jiggle" function). This ensures that pairs of errors are split up and redistributed through the string as a whole. Alice and Bob also choose a new block size, twice that of the previous block size. As there are now fewer errors left to correct (some having been hopefully removed in the first round), the block sizes do not need to be as small to minimise the pairings of errors.

Alice and Bob then repeat the process of exchanging parities and correcting errors. However, in any pass of the protocol other than the first, each error corrected can then be used to correct additional errors. This cascading of error correction is where the protocol got its name. It works as follows. If you've corrected an error in the second pass of the protocol, it is possible to retrace that bit's path through the protocol and find which block it was in in the first pass. As that block passed through the first pass unharmed, it must therefore contain an even number of errors, i.e. one additional error that you haven't found yet. You can then exchange all the relevant parities for that block until you find the second error.

If Alice and Bob entered cascade with $p \leq 0.15$, then running four passes of cascade should eliminate all their errors.

2.3.4 LDPC codes

LDPC codes have now succeeded the cascade protocol, mostly due the huge inefficiency of the number of communications required to use it.
At first, LDPC codes were even more inefficient than cascade [104]. Additionally, they only really work for a constant, known information rate without major

modifications [105, 106].

LDPC codes are best visualised with a graph (figure 2.7). The graph describes a series of conditions which a codeword (the output of your encoding mechanism) has to satisfy. The top row of boxes represent each digit of the codeword. Each of these top boxes feeds, via a line, into a number of bottom boxes. The sum modulo 2 of the inputs to each of the bottom boxes must be zero.

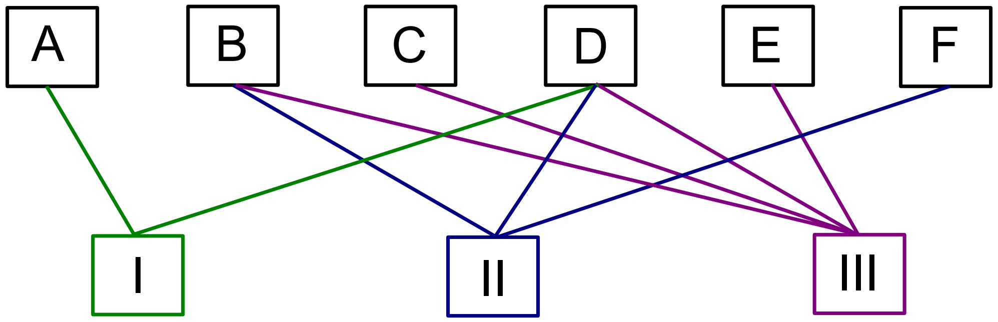


Figure 2.7: A graph of an LDPC code.

This can then be represented in a parity check matrix (equation 2.14), where each vertical column represents each position in the codeword, or each of the top boxes in the diagram. Each horizontal row is the set of parity check constraints, or the bottom boxes. Whether each matrix element is a one or a zero is dependent on if the top box is connected to the bottom box by a line.

$$H = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 \end{pmatrix} \tag{2.14}$$

After some elementary row operations (swapping and addition modulo 2), the parity check matrix can be written in the form $H_1 = [\mathbf{P}^\top | \mathbf{I}_{n-k}]$, where $\mathbf{I}$ is the identity matrix, n the length of the code word (number of top boxes), and k the size of the input (number of bottom boxes).

$$H_1 = \left(\begin{array}{ccc|ccc} 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 \end{array}\right) \tag{2.15}$$

This can be used to create a generator matrix, $G = [\mathbf{I}_k|\mathbf{P}]$ which can be used to produce a codeword for each of the eight possible 3 bit long input bit strings.

$$G = \left(\begin{array}{ccc|ccc} 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \end{array}\right) \tag{2.16}$$

$$\begin{pmatrix} 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 0 \end{pmatrix} \tag{2.17}$$

The input bit string is the first three bits of the output codeword.

In our reconciliation scenario, Alice sends this codeword to Bob. Bob first multiplies the received codeword by the original parity check matrix, H. If the codeword is correct, the output will be zero. If the output is not zero, then Bob utilises a decoding method called "belief propagation". Here he uses the constraints from the original graph and check matrix to logically deduce the position of the error. For example, if Bob receives $\begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}$ then he can multiply this with the check matrix, and get out

$$\begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \tag{2.18}$$

Which has a one in it, so he knows that an error must have occurred during the classical transmission of that codeword.

He then uses a logical sequence of steps, and the known error rate of the channel to fix the error[1]. This is easiest to visualise with figure 2.8.

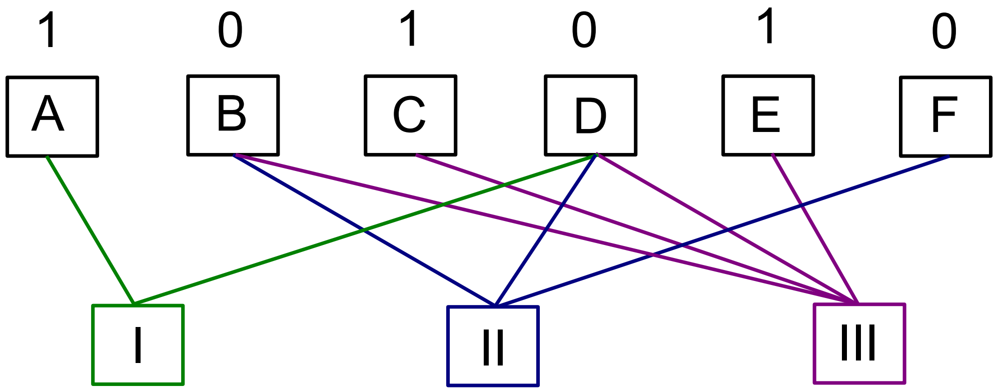


Figure 2.8: An LDPC graph with a codeword containing an error.

Firstly, Bob looks at the first check node, box I. He finds that the sum (modulo 2) of the input nodes (A and D) is one. This means that either symbol A or symbol D of the codeword is an error, as the parity check boxes should add to zero. He flips the value of symbol A, and checks the validity of the new codeword by multiplying it by the check matrix H, which outputs zeroes and tells him he has now got a correct copy of the codeword Alice sent.

If there was more than one error, things get really logical and awesome. Take for example, if Bob had received $\begin{pmatrix}0 & 0 & 1 & 1 & 1 & 1\end{pmatrix}$. Putting this into the check matrix would have given him $\begin{pmatrix}1 & 0 & 1\end{pmatrix}$. Breaking this down further (either via the graph which is easy to visualise, or the matrix if you like maths), the parities are shown in figure 2.9.

If the error rate was assumed to be one error per block, then Bob starts out by looking at which check nodes have odd number of errors in, in this case, nodes I and III. The common codeword node between these two is D. Bob assumes (or "believes") that this is an error. If this is the case then he can propagate this through, to know that there must also be an additional error in check node II.

[1] you might think this is going a bit deep with the description here, but hold on, it comes in handy later.

$$\begin{array}{rll}
\mathrm{I}: & \mathrm{A} \oplus \mathrm{D} = 0 \oplus 1 = 1 & \text{Error} \\
\mathrm{II}: & \mathrm{B} \oplus \mathrm{D} \oplus \mathrm{F} = 0 \oplus 1 \oplus 1 = 0 & \text{No/Even errors} \\
\mathrm{III}: & \mathrm{B} \oplus \mathrm{C} \oplus \mathrm{D} \oplus \mathrm{E} = 0 \oplus 1 \oplus 1 \oplus 1 = 1 & \text{Error}
\end{array}$$

Figure 2.9: Parity checking for a codeword of 001111.

As, if D was the only error in this check node, then it would have produced a parity of one. The only codeword node in this check node and none of the others is F. Therefore F must also be an error.

Most people use these as error correcting code in this way. However, for Bob to spot errors in the quantum transmission (where both Alice and Bob will be producing LDPC codewords which are valid with the parity check matrix), this will not work. Instead they have to simply compare their codewords in a normal manner.

Some security is maintained either through the clever choice of LDPC codes (there are a vast range of possible codes to use, and choosing one well suited to your application is vital. There are a number of rules for their construction), or through the use of puncturing [107]. This is the practice of removing one of the symbols of the code word before sending it, making decoding a bit more tricky.

Other methods also exist, such as sending the first set of digits, the original input bits, (in our example the first three) through the quantum channel, and sending the second set of digits, the check bits, through the classical one, to be recombined at Bob's end [108].

Other error correction protocols such as multi-dimensional reconciliation algorithms (MDRA) [109] have also been suggested, but not widely used.

2.3.5 Privacy amplification

Mentioned here for completeness, privacy amplification [108, 110, 111] is very simply the act of using hash functions to try and increase Alice and Bob's information advantage over the eavesdropper.

Hash functions are functions which can take data of any size as an input and output a fixed length string. Cryptographic hash functions are designed to be incredibly difficult to reverse, and with few collisions (two inputs producing the same output).

Chapter 3

Thermal state protocol

3.1 Introduction

In this chapter we introduce a novel protocol for continuous variable quantum key distribution using thermal states. This is the first version of this protocol, and here serves mainly as a rough draft of where we wish to be in the future. Each element of this protocol will need testing and analysing both by itself, and as part of a whole, before becoming complete. We take the first steps towards doing that in this thesis. It is extremely probable that during the process of examining the individual stages, problems will be found which will necessitate changing parts of the protocol. This is to be expected, and should indeed happen in order for the final protocol to be robust.

3.2 Protocol

1. Alice creates a beam from a trusted thermal source.

2. Alice applies an amplitude modulation to the beam using values drawn from a Gaussian distribution.

3. Alice uses a trusted beamsplitter to divert a portion of the beam into her own detector.

4. The remainder of the beam is transmitted along an AWGN (additive white Gaussian noise) channel to Bob.
5. Bob measures his received portion of the beam.
6. Both parties then discretise their measurement results from continuous variables to a bit string.
7. Bob chooses a random selection of his data to send to Alice over an authenticated public channel, so that they can estimate their error rate.
8. Alice and Bob then perform advantage distillation to try and reduce the potential knowledge of an eavesdropper.
9. This is followed by a classical reconciliation (error correction) procedure Cascade
10. Privacy Amplification is then used to ensure any eavesdropper has minimal knowledge of the key.

3.3 Discussion

In our protocol, Alice uses a thermal source to produce her beam, rather than a coherent source such a a laser. In chapter 5, we examine how the extra correlations present in a thermal source may alter the security of a transmission. Alice then applies a modulation to the beam, before taking a portion of it to measure herself. The remainder of the beam is sent along a channel to Bob and Eve. After the transmission, Alice and Bob perform several common classical reconciliation steps in order to obtain a secure, error free key. As stated above, this is an idealised protocol in the first stages of development, and will doubtless need significant improvement before being secure. Due to the possible increases in security by the use of thermal states, we feel that this is an avenue well worth exploring.

Chapter 4

Laser Characterisation

4.1 Lasers

Lasers consist of a cavity containing a gain medium, with mirrors at each end. One of these mirrors is only partially reflective, allowing a small portion of the light to escape and form the laser beam. The rest of the light remains inside the cavity, continuing to bounce off the walls either end.

Our system is an external[1] cavity laser, which means that the cavity extends beyond the gain medium. In this case, the gain chip forms the gain medium, the back of which also forms one of the reflective edges. Once the light has left the gain chip, it travels through free space until meeting a refraction grating where it is then passed to a mirror. This mirror then reflects the light back over the diffraction grating and into the gain chip. The advantage of this set up, is that the light has to pass over the diffraction grating twice in each pass, resulting in a much smaller linewidth than a conventional cavity laser.

A gain medium is a substance full of atoms with two energy levels. Most of the time, these atoms are in the ground state of energy, but when you turn the laser on, a population inversion is created, meaning that most of the atoms are now in the excited state. These atoms have two ways of decaying, “spontaneous

[1] Sometimes extended cavity laser

emission" and "stimulated emission". During spontaneous emission, atoms are caused to decay due to vacuum noise, releasing a photon in any available mode. During stimulated emission, an initial photon will cause the atom to decay, releasing an additional photon in the same mode as the original. (Same direction, phase, amplitude, etc.)

There are several different types of gain medium. The first lasers used ruby, and modern lasers use anything from gasses to dyes.

In our case, we use a semiconductor diode. This consists of a p-region made from a material with excess holes (too few electrons) and an n-region made from a material with an excess of electrons. Between these is the i- (or interaction) region. When a current is passed through this sandwich of materials, the electrons and holes both move into the interaction region where they can recombine and produce photons via spontaneous emission. If the current is high enough, an excess of electrons and holes enter the i-region, and if the rate of spontaneous emission is not fast enough, you end up with a population inversion. All this now takes is a single photon to then trigger an avalanche of photons via stimulated emission.

4.1.1 Super luminescent diodes

A light emitting diode (LED) contains an almost identical set-up, the difference being that an LED relies solely on spontaneous emission, whereas a laser relies on stimulated. It is almost impossible to remove all spontaneous emission from a laser, and good laser design maximises the proportion of stimulated emission while minimising the spontaneous.

Light produced via spontaneous emission is sometimes referred to as luminescence. Super luminescent diodes, or super luminescent LED's (SLED's) emit light via a process known as amplified spontaneous emission (ASE). Again, this takes the same sort of hardware as LED's and lasers. In this case, the light is produced by spontaneous emission, and then further amplified by stimulated emission. You would be forgiven for thinking that this sounds exactly like a laser. Because it does. However, the difference here is the lack of feedback. In a laser, in order to produce light in a single mode, you need a large amount of feedback, a lot of reflection from the mirrors at either end of the cavity. This ensures that as soon as one mode becomes dominant, it stays dominant and becomes increasingly dominant as more and more photons from that mode trigger more and more avalanches of stimulated emission.

However, in SLED's where the feedback is much lower, no single mode will dominate. Instead photons from different modes each trigger smaller scale avalanches, resulting in no overall domination by any single mode.

In our case, the same gain chip, and cavity, is used for both ASE and lasing. Here however, the switch between the two forms of emission is governed by the size of the population inversion, which in turn is decided by the size of the driving current. With a small current, and small population inversion, the size of the avalanches are limited, meaning that no one mode can dominate. This produces thermal ASE luminescence. However, when the current, and population inversion are increased, the large number of potential electron hole pairings mean that sizeable avalanches can occur, large enough for one mode to win out. Meaning that the system is producing coherent laser light.

4.2 Experiment

For this part of the experiment, the main focus was to characterise the light produced by the source, and to determine the existence and position of the threshold between the production of thermal and laser light.

We start with a brief overview of the equipment used.

4.2.1 Equipment

Laser

A ThorLabs tunable laser kit at 770nm (TLK-L780M) was chosen as the source of both thermal and laser light. This was due to its compatibility with existing lab components, and its adaptability. The kit consists of a diode chip, along with all the usual laser components, collimating lenses, a mirror and a diffraction grating set up in a Littman configuration[1]. Each element of the laser can be swapped in or out, as well as being tuned, to enable a high degree of adaptability.

The Litmann configuration of the device, as shown in figure 4.1 meant that light from the gain chip was passed through a collimation lens, before hitting a diffraction grating. The 0th order reflections from the grating were coupled to a mirror, which reflects the light back onto the grating. This double diffraction enables a much smaller linewidth. The first order reflections in one direction are coupled back into the gain chip, while those in the other form the output.

In order to set this up to produce both thermal and coherent light with minimal differences between the two, we started with a Fabry-Perot gain chip (TFP780A), with a centre wavelength of 770nm when lasing, and 780nm during the ASE regime. We changed the tuning of the external cavity so that the lasing output was also at 780nm. The wavelength is changed by moving the mirror back and forth to change the length of the external cavity.

We chose our wavelength by plugging the tuning motor into the PC and monitoring the output with an optical spectrum analyser (OSA). This ensured that

[1] There are two types of tuneable external cavity laser, Littman and Littrow. The Littrow configuration is simpler, as it omits the mirror, and the output is the 0^{th} order of the diffraction grating. The wavelength is changed by rotating the angle of the diffraction grating. Unfortunately, this means the output direction of the beam moves as well. The Littman configuration, as we see here, has two advantages over the Littrow. The output beam direction remains constant, and the two uses of the diffraction grating ensure a tighter linewidth. The disadvantage is that you're using the 1^{st} order diffraction from the grating as the output beam, so the output power is lower.

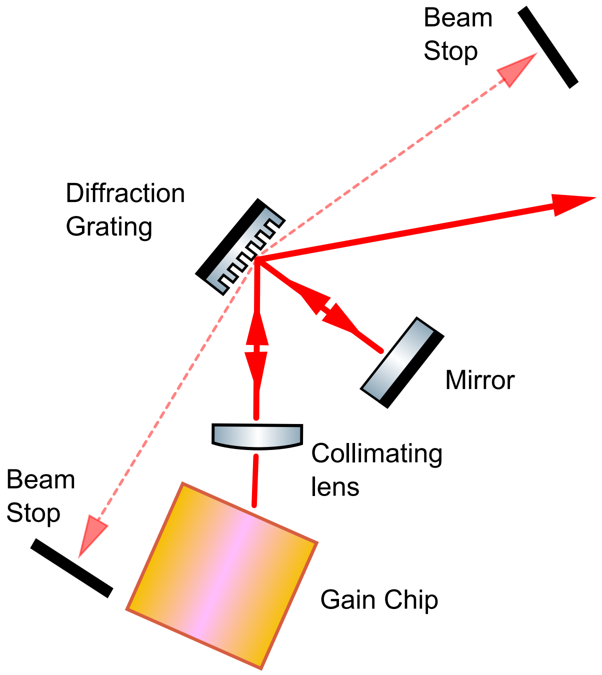


Figure 4.1: The Litmann configuration of an external cavity laser. Light produced in the gain chip is formed into a column via the collimating lens, where it is then reflected off the diffraction grating onto a mirror. This mirror marks the far end of the cavity. From here, light passes back off the diffraction grating and into the gain chip. The first order diffraction in the opposite direction becomes the output laser beam.

the wavelength stayed constant through the ASE regime into the lasing. This tuning motor alters the distance of the mirror from the grating. Then we had to get rid of mode hops using the mode hop adjustment knob. This is a lot easier as we are keeping a stable wavelength rather than scanning through a range. This time we had to connect the output to a scanning Fabry-Perot Etalon which was then connected to the oscilloscope in order to view the output. After that, it was a simple matter of adjusting the mode hop adjuster knob until the mode hops vanished.

Finally, it was necessary to fine-tune the source, using the focus adjuster and the flexure adjustment screw to maximise the output power of the laser. This is can be challenging as the two are not independent of each other. No part of the kit is properly independent from any of the other parts, but this one is the most tricky. The focus adjuster moves the collimating lens towards or away from the gain chip, and the flexure adjustment screw adjusts the tilt of the grating.

Drift occurs in the laser and this last step needs to be repeated approximately every two weeks in order to prevent the output power from falling too much.

The laser itself is connected to an integrated laser diode and TEC (thermo-electric cooler) controller. The TEC module ensures that the insides of the gain chip remain at a constant temperature (in this case 25^{o}C), while the laser diode controller allows us to adjust the current through the diode. This was kept in constant current mode (rather than constant power mode where the laser output power is kept constant) as it is the current level which interests us.

Detectors

The detectors used were ThorLabs Det36A. These are battery operated and have an advertised rise time of 14ns. These consist of a silicon diode which operates in much the same way as any other diode, but when light is absorbed by the depleted region, a photocurrent is generated.

Dark current is current that leaks across the diode due to the reverse bias applied across it. This is always present, and is usually reasonably small. In our case, about 0.35nA.

The bandwidth and response times of the photodetector are governed by a few main factors. Drift time, is the time it takes electron-hole pairs to move from the depletion zone to the electrodes. This is usually of the order of ps. Diffusion time, is the time it takes electron-hole pairs generated outside the depletion zone to reach the electrodes. This is slower than the drift time, and the proportion of these produced is wavelength dependent. The rise time of the detector is dependent on the junction capacitance, and the load resistance. In our case the load resistance is 50Ω as we connect the detector with a 50Ω coaxial cable to the oscilloscope which terminates it at 50Ω.

Unfortunately, due to the response times varying with wavelength, and the advertised rise time of 14ns being given for 632nm, as we are using a wavelength of 780nm, which is towards the near infra-red, we can expect somewhat higher rise times.

Oscilloscope

The oscilloscope used in this experiment for reading the output from the detector was a LeCroy Waverunner 44xi. We had originally intended to use a National Instruments PCI-6034E data acquisition device (DAQ), but found that it was too slow. The LeCroy was chosen for its speed. The rise-times of the 44xi are significantly shorter than the integration times of the detector, and as such should not affect the data acquisition.

There are three main things we need to know about any oscilloscope. The first is the bandwidth. In the case of oscilloscopes, this is defined as the frequency at which the recorded amplitude of a sine wave drops by 3dB. If you input a sine wave of fixed amplitude into the scope, then as you increase the frequency of that sine wave, the limited bandwidth of the scope causes the recorded amplitude to drop. Purely because it can no longer keep up with how fast things are going. Once the amplitude has dropped by 3dB, that frequency is defined as the bandwidth of the scope. The bandwidth of the 44xi is 400MHz.

The second thing we need to know is the rise time. This is defined as the time it takes to get from the minimum to the maximum input. For example, if a square wave was input into the scope, it would be displayed with a slightly sloped edge and rounded corners, rather than with nice straight lines and sharp right angles. Again, this is because the scope is too slow to observe the sharpness of the transition. The rise time of this scope is 875ps[1].

The third important thing is the sample rate, which gives the number of samples the scope takes each second. In this case, 5GS/s (5×10^9 Hz). This is useful for us to know because it forms the time interval τ later on.

Optical Isolator

Optical isolators are also known as optical diodes, as they only allow light to pass in one direction. They are used to prevent stray reflections from experimental components re-entering the laser and causing damage.

Their principle of operation is fairly simple, they consist of three main components. Firstly is a horizontal polarising filter. This is followed by a Faraday rotator. The Faraday rotator uses a magnetic field to rotate the polarisation of the light through 45^o. The final component is a second polariser, at 45^o to the first one, enabling the light to pass through. Any light attempting to pass through in a backwards direction will enter the isolator at 45^o through the second polariser. It's polarisation is then rotated an additional 45^o by the Faraday

[1] If you check this in the manual, you will find it reads 875ns. This is a typo. Oscilloscopes almost universally have bandwidth × rise time ≈ 0.35. All of the other oscilloscopes in this range follow this rule.

isolator, meaning it is now in a horizontal polarisation. No component of it can through the first, vertical, polarising filter.

Rotating half wave plates

These consist of a half wave plate mounted on a computer controlled rotation mount. This is controlled using Thorlabs ATP$^{\mathrm{TM}}$ and later, Kenesis® software. For experiments in chapter 5, it was controlled directly from a python script using PyAPT commands [112].

Half wave plates consist of a birefringent crystal, through which linearly polarised light will pass through faster in one direction than another. This causes the phase of each component of the polarisation to change during its transit through the crystal, and thus the overall polarisation of the light is different when it emerges from the other side.

The half wave plate causes the polarisation of the light to rotate through an angle of 2θ, where θ is the angle between the fast axis of the half wave plate and the incident polarisation direction.

In these experiments, the half wave plates are often coupled with polarising beam splitters, which cause light components polarised in the vertical direction to go one way, and the horizontally polarised components to go another. The combination of these two components acts as a variable beamsplitter.

We are mostly concerned at this stage with the viability of using the current to switch the laser[1] seamlessly between producing thermal and coherent light. To this end, we begin characterising the output of the laser.

[1] To aid confusion, the light source is referred to throughout as a "laser", whether it is lasing and producing coherent light, *OR* producing thermal light in the ASE regime.

4.2.2 Wavelength spectrum

First, the equipment was set up as shown in figure 4.2, to examine the wavelength profiles of the beam above and below the thermal-coherent threshold. The beam is passed through an optical isolator to prevent stray reflections from damaging the laser, and then passed onto a pair of steering mirrors to align the beam into the fibre. The fibre then fed directly into the optical spectrum analyser (OSA). The OSA was connected to a PC, where the spectrum was recorded for each operating current, and the results saved to a portable USB disk for transfer to another computer to construct the graphs from the CSV files.

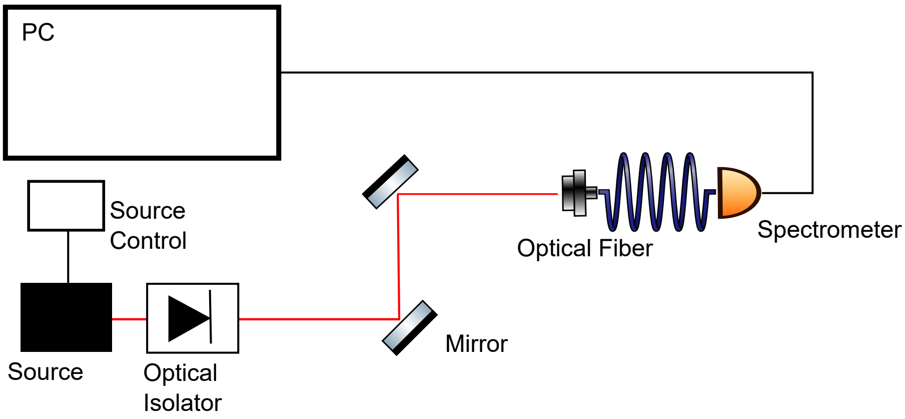


Figure 4.2: The beam from the source is passed across a pair of steering mirrors and into a fibre couple which links to the optical spectrum analyser, enabling us to measure the wavelength of the source.

Figure 4.3 shows the output of the OSA at every 10 mA increase in source current. The OSA software applies some normalisation to the intensity, so that the higher intensities approach one. Increasing the source current increases the intensity.

Figures 4.4a and 4.4b show this same data, but as the relative intensity and the FWHM respectively as a function of the current. There is an interesting

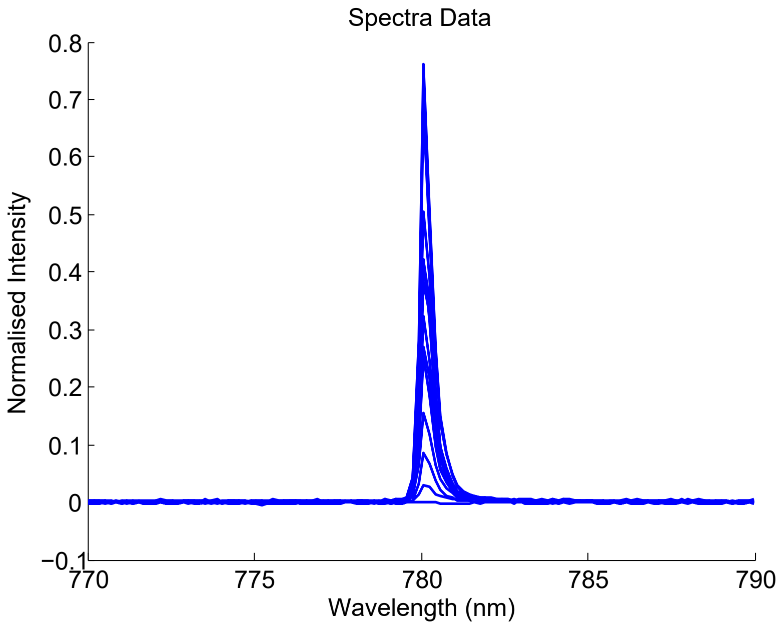


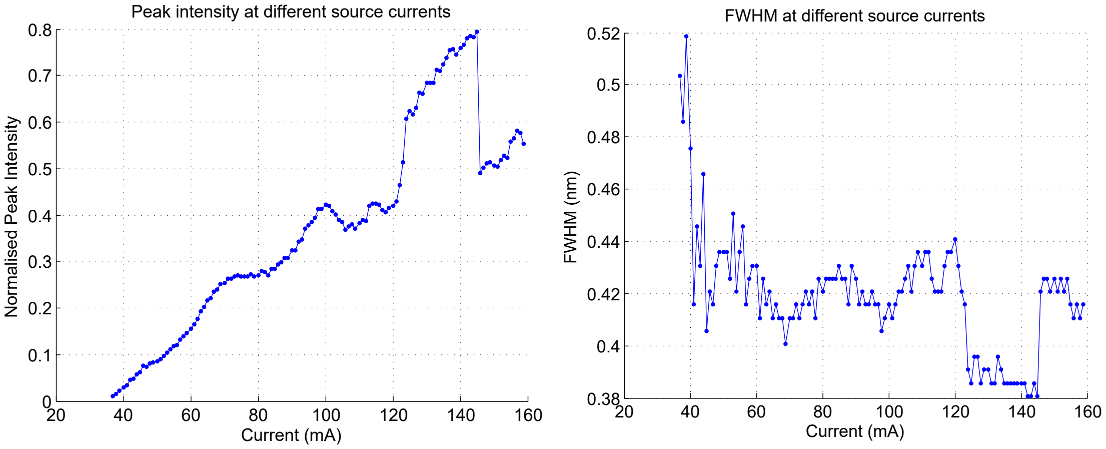
Figure 4.3: The spectral data of the source, beginning at 30mA (flat line) and increasing in 10mA steps until 150mA (peaked lines).

correlation between the two figures. Where the intensity peaks, the FWHM gets narrower.

The FWHM is relatively noisy, which is at least partly due to its method of calculation. The wavelength precision on the OSA meant that relatively few data points were used to construct the peak. This meant that instead of a nice smooth curve to work with, we instead had a rather lumpy one, losing some of the natural shape. Details of this calculation can be found in the supplementary CD.

Another example of the OSA precision can be found looking at the peak wavelength, shown in figure 4.5. While you would expect small natural deviations in the peak wavelength, none are shown here. This is due to the fact that any small deviations are lost in the lower precision of the instrument.

Fortunately, as we are only concerned with rough outcomes at this stage, the OSA precision is not a huge obstacle.



(a) Relative peak intensity as a function of source current

(b) FWHM as a function of source current

Figure 4.4: Peak intensity and wavelength full width at half-maximum for different source currents.

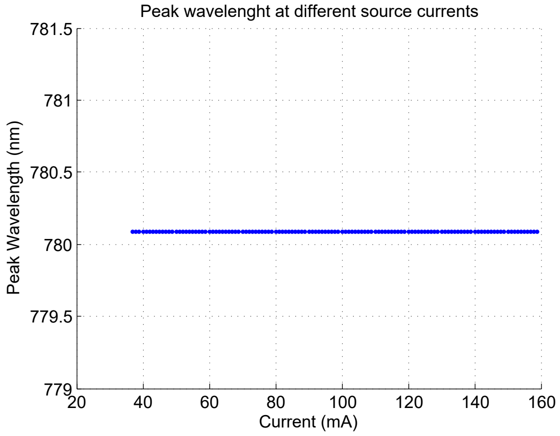


Figure 4.5: The wavelength at the peak intensity as a function of source current.

4.2.3 Coherence Time and the expected $g^{(2)}(0)$ peak height

A single trace from the OSA is shown in figure 4.6. From this, we can find the peak wavelength λ_0, and the FWHM $\Delta\lambda$. With the detector integration time, these enable us to calculate the expected $g^{(2)}(0)$ peak height in our system via equation 1.57.

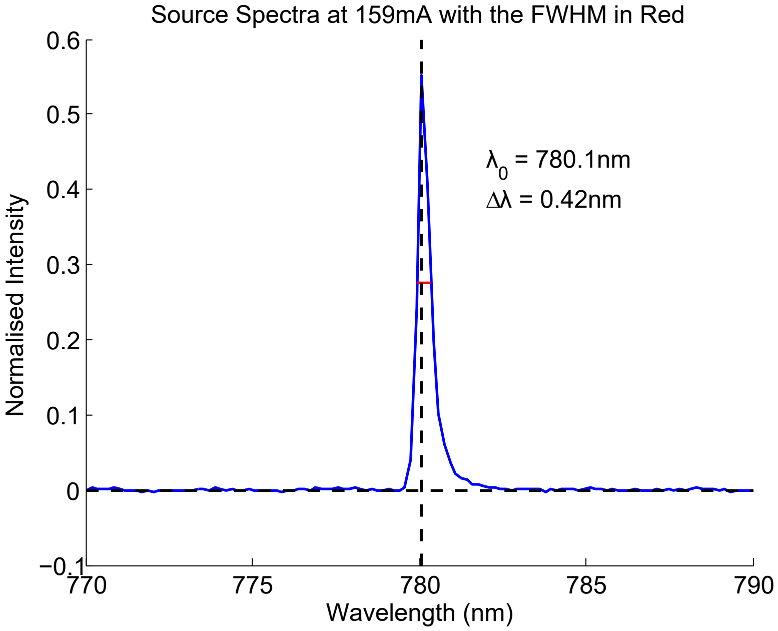


Figure 4.6: Spectral data at 59mA.

While it is clear from figure 4.5 that the wavelength remains stationary at $\lambda_0 = 780.09$nm, $\Delta\lambda$ varies between about 0.40 and 0.44nm over the majority of the range. As we are only interested in a rough idea of how high we should expect our $g^{(2)}$ peak to be, we will take $\Delta\lambda = 0.42$nm. This gives our light a coherence time of

$$\tau_c = \frac{\lambda_0^2}{c\Delta\lambda} = 4.8\,\text{ps} \tag{4.1}$$

We know that the rise time of the oscilloscope is 875ps, and that the rise time of the detector is more than 14ns. Unfortunately, the equation we are using is very sensitive to the response time, but as we are only after a rough estimate, we will use $T_R = 14\text{ns}$.

These values of τ_c and T_R when substituted into equation 1.57 give us an answer of $g^{(2)}(0) = 1.0003$ which is significantly below 2. The expected longer detector response times will give an even smaller value of $g^{(2)}(0)$.

4.2.4 Thermal-coherent threshold

Below an operating current of about 124mA, the laser emits thermal light via ASE, as there is not enough feedback to meet the lasing conditions. However, above this level the laser lases as you would expect, emitting coherent light.

To examine this threshold, we used an experimental set-up based on that described in section 1.1.3. Light from the source is passed through an optical isolator, before being split equally between the two detectors using a combination of a rotational half-wave plate and a polarising beam splitter. This combination of components allows an easy way to control the portion of light received by each detector. Here, of course, we are only concerned with a 50:50 split, so a non-polarising beam splitter would have been sufficient, but this arrangement allows us greater flexibility further on with minimal disturbance to the system.

The half-wave plate was controlled using the Thorlabs APT™ software towards the beginning of the project, and graduated to the Kinesis® software towards the end, due to the purchase of additional equipment and the necessary upgrading of both software and computer system[1].

The light finally hits the detectors, the signal of which is recorded by the oscilloscope and saved onto a USB drive. This is then transferred to a computer where the data analysis takes place. This set-up can be seen in figure 4.7.

At the beginning of the experiment, the laser was first turned to 160 mA, where the alignment of the system, and the equal division of light through the

[1] Surprisingly few things in this world are backwards compatible.

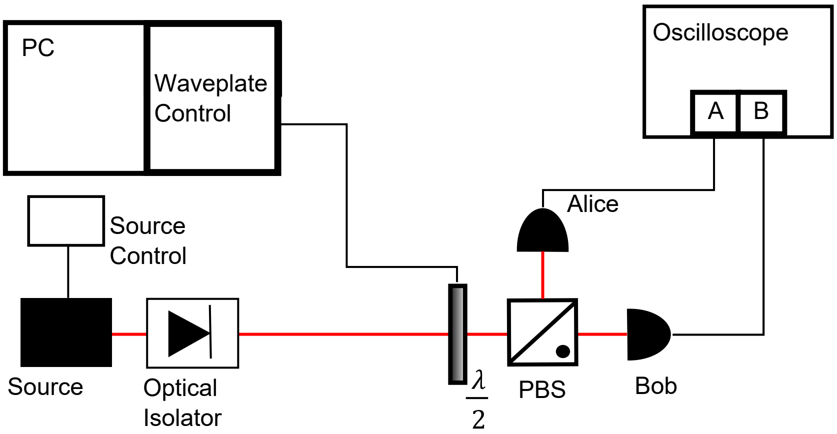


Figure 4.7: Light from the source is split equally between the two detectors by means of a rotatable half-wave plate ($\frac{\lambda}{2}$) and a polarising beam splitter (PBS). Data from the detectors was recorded by the oscilloscope, and then moved to a computer where the correlation was calculated.

beam splitter were checked.

The light source was supplied with a current of 30mA to begin with, well below the advised operating current of 140 mA. For each run of the experiment, the current was increased by 1 mA up until a final value of 160mA[1]. During each run, data was recorded on the oscilloscope for both detectors simultaneously. A 5μs data sample was taken at 5GS/s. To obtain the highest level of detail about the signal, the scope was kept at its smallest voltage range of 10mV/div throughout the entire experiment, with the voltage offset being adjusted where necessary.

Once collected, the data was transferred to a PC to perform the analysis. For each trace, the raw data was plotted, and the correlation[2] calculated. This was done as follows.

[1] Occasional values are missing due to human error. It was judged to take longer writing code to automate this process than it would to simply do it.

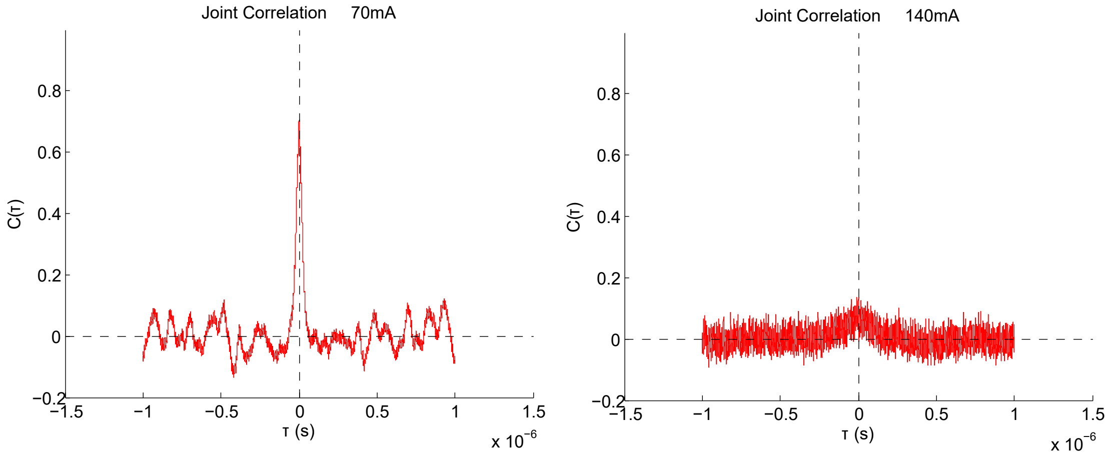
[2] See section 1.1.4

First, the mean was subtracted from each data set so that they were normalised around zero. Then the two data sets were shifted relative to each other, by one time step $\Delta\tau$ each time. $\Delta\tau$ is the difference in time between each data point recorded by the oscilloscope, and in this case, $\Delta\tau = 0.2$ns. Of course, when $\tau \neq 0$, the edges of the data overhang each other, and don't have a corresponding data point in the other dataset. There are a number of ways of dealing with this. Padding the data set with zeroes to keep them the same length and then calculating the correlation will give a triangular pattern, more difficult to interpret. Another option is to trim the excess of each data set, and work with a variable sized portion. This again produces interesting effects around the edges. We opted for using the same sized portion of data to correlate throughout, by comparing one fifth of the length (5000 data points, or 1μs of data) at each step. We used the whole dataset for the normalisation.

For each step, the correlation was found via equation 1.37. Each point in the section of data we were looking at was multiplied by its corresponding point in the other dataset, and then the mean was found. This was normalised by dividing by the root mean squared (RMS) of both data sets. This squaring and then square rooting is necessary due to the fact that the data had been normalised so that the mean was zero.

There are clear differences in the graphs of the correlation as a function of τ, the time shift between the two detectors, as the source current is increased. For example, figure 4.8a shows the correlation at 70mA, which is below the recommended operating current of the source, and in the ASE regime, whereas figure 4.8b shows the correlation at 140mA, which is the recommended operating current.

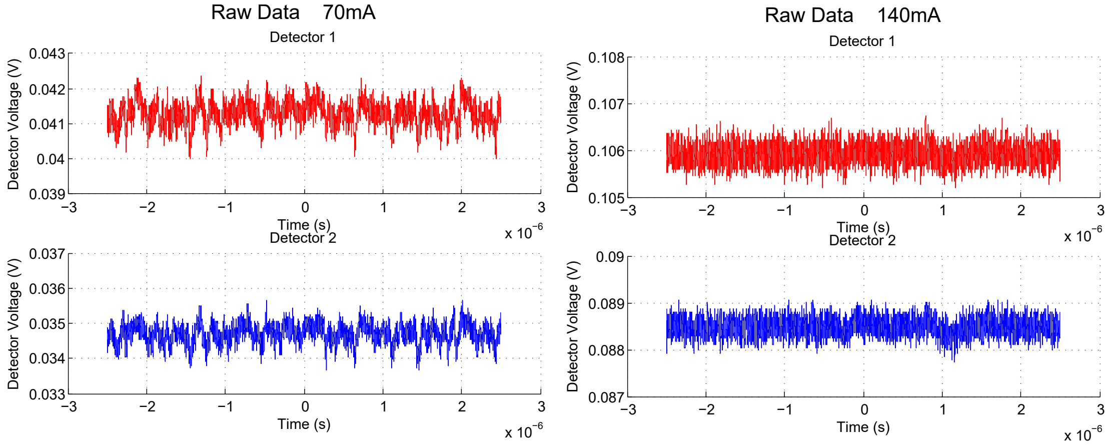
The graph at 70mA clearly shows the existence of correlations absent in the 140mA regime. These correlations arise due to fluctuations in the light field which can be seen in the raw data, figure 4.9.



(a) Correlation at 70mA

(b) Correlation at 140mA

Figure 4.8: Correlation of the outputs of two detectors when the source current is at 70mA (thermal) and 140mA (coherent).

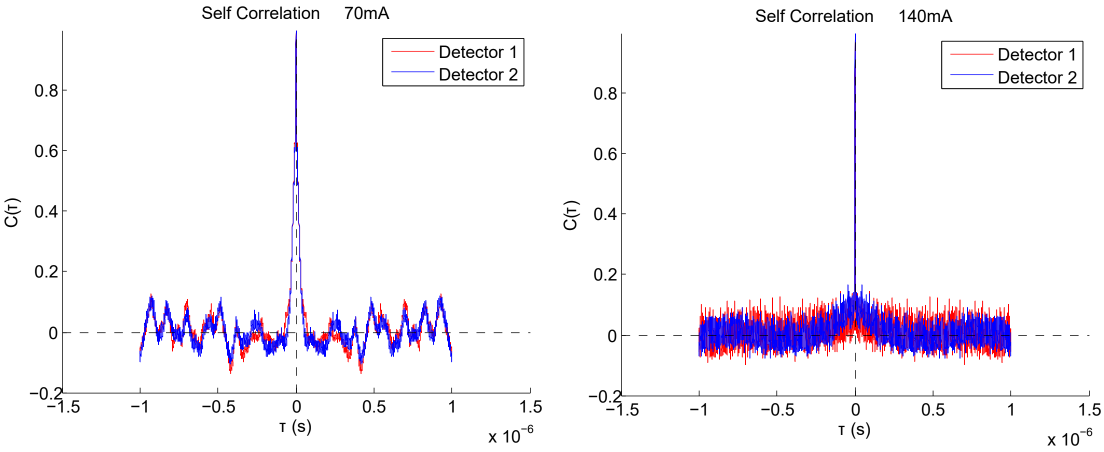


(a) Raw Data at 70mA

(b) Raw Data at 140mA

Figure 4.9: Detector voltage outputs for the source currents of 70mA and 140mA.

As you would then expect, the correlations arising from these fluctuations can also be observed in the self correlation of each detector with itself. This is simply a matter of duplicating the output of a single detector, and comparing the original output with the duplicate shifted in time. These are shown in figure 4.10. The self correlation at 140mA shows a peak not present in the joint correlation at this current because when correlating with yourself, you will always be correlated when $\tau = 0$, there is no shift between the two data sets.



(a) Self correlation at 70mA

(b) Self correlation at 140mA

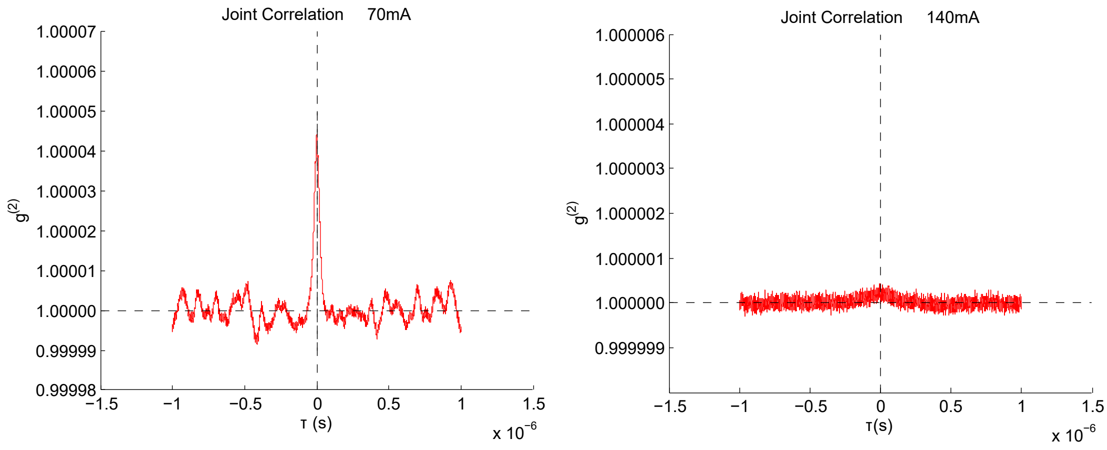
Figure 4.10: Self correlation of detector outputs at source currents of 70mA and 140mA.

You may have noted that so far, we have been using the correlation function rather than the $g^{(2)}$. This is because for our purposes, we are only concerned about the existence of correlations, rather than being overly concerned by their magnitude. In addition, as we mentioned in section 1.1.6, due to the speed of our detectors, we would be working with incredibly small values of $g^{(2)}$. Indeed, as you can see from figure 4.11, the $g^{(2)}(0)$ we obtain is actually an order of magnitude smaller than we had estimated at 0.00004, instead of 0.0003. This additional drop is likely to be due to the placing of the detectors. We calculated the coherence time of the light (equation 4.1) to be approximately 5ps. The most convenient measure of the speed of light is 1 foot per nanosecond[1]. This means that for both

[1] Another good unit is a millifortnight. 20 minutes.

detectors to be within the coherence time of the light, the distance between each detector and the beam splitter had to be no more than 1.5mm different. We did our best, but trying to juggle alignment and distance is challenging. It is likely we were a little way out, but not too far out, as we do still see the correlations.

One way in which to improve upon this would be to use the correlations themselves to optimise the detector placement. This would involve mounting the detectors onto a moveable stage and adjusting their position to maximise the correlations. This would require a large amount of both time and experimental skill as the stages would have to be angled correctly and in the correct horizontal orientation before the depth could be altered.



(a) $g^{(2)}$ 70mA

(b) $g^{(2)}$ 140mA

Figure 4.11: $g^{(2)}$ correlation of detector outputs at source currents of 70mA and 140mA. Note the change in scale between the two figures.

The height of the peak (either in $g^{(2)}$ or the detector correlations) will be used as a measure of the amount of correlations, and thus the degree of thermality of the light. As we are interested in how the structure of the light changes with the operating current of the laser, we plot the height of the correlation peak against the source current in figure 4.12. We see that the correlations drop dramatically from 105mA onwards, indicating the coherent lasing regime, with a brief return to thermality between 117mA and 123mA.

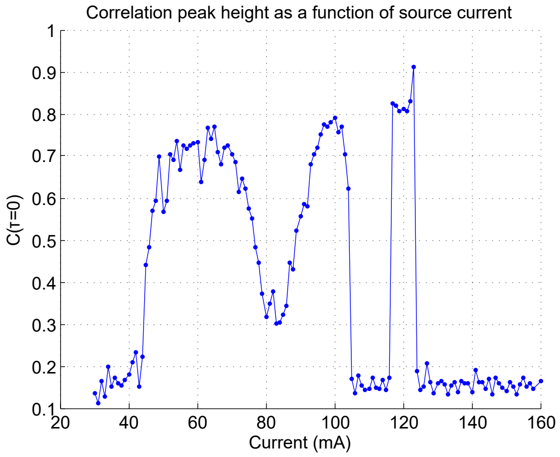
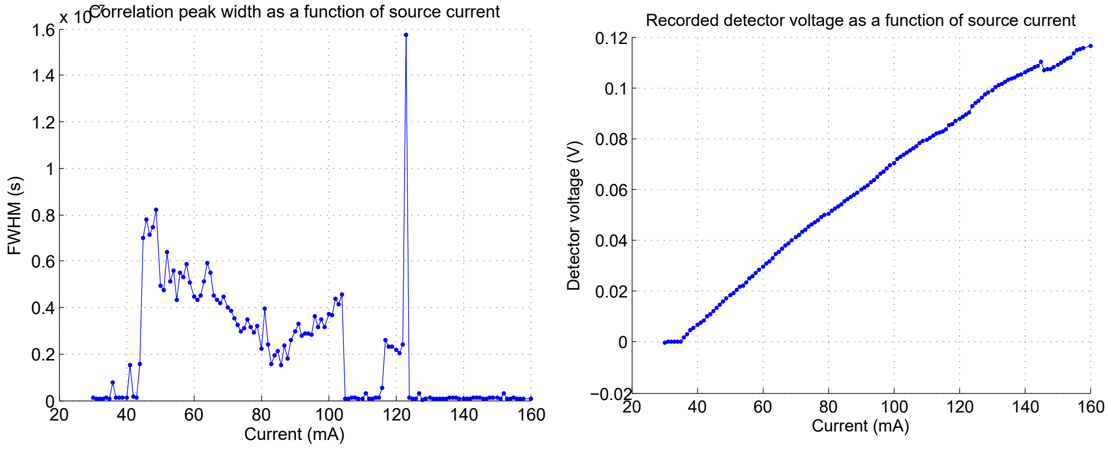


Figure 4.12: Correlation height at $\tau = 0$ as a function of source current

We also plotted the FWHM of the correlation peaks, and mean detector voltage output as a function of the source current, (figure 4.13) to see if there were any conclusions which could be drawn from those. There is a clear correlation between peak height and width, which is not particularly unexpected, but the main point of interest lies in comparing these graphs to those from the OSA in figure 4.4. The same intensity characteristics can be seen in both the detector voltage and the OSA intensity. This is good, it would be a bit worrying otherwise. And while the intensity peaks of fig. 4.4a match the dips in $\Delta\lambda$ (fig. 4.4b), these occur after 123mA, rather than prior as seen in the correlation graphs.

Of course, these graphs are drawn with only one run of the data, and give no indication as to if, and if so how much, variation there is in these points. To examine this, we repeated the experiment over the interesting set of currents (60-130mA) in two different ways. First, we ran the range of currents twice more without turning the source off in-between each run (figure 4.14a), and secondly, we did the same, but turning the source off and on again each time (figure 4.14b). While the source is kept on, there is very little variation in the peak height, but when the source is turned off between runs, the large transitions towards the



(a) Correlation peak width

(b) detector voltage output

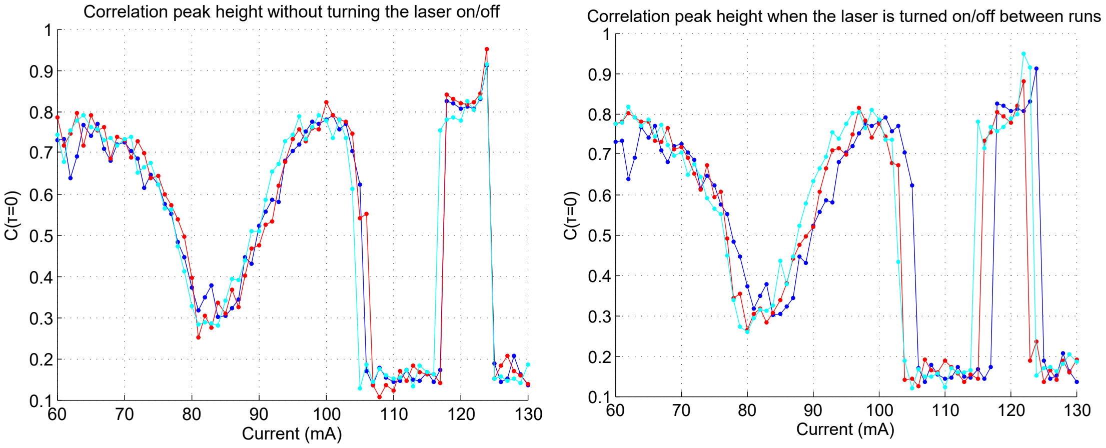
Figure 4.13: Additional measures as a function of source current

higher end of the current range shift by as much as 3mA. This isn't a problem, it's just useful to be aware of when these things happen.

The temperature of the gain chip in the source is controlled via the thermoelectric cooling (TEC) system. This keeps the temperature constant to within ±0.01^{o}C. The temperature of this chip has a strong effect on the functioning of the source, which is why it is controlled so closely. To see what would happen, we ran the same experiment again, but with the chip at 20^{o}C rather than the recommended 25^{o}C. This is shown in figure 4.15. The pattern of transitions remains broadly the same, only with less sharp transitions.

Now, there is an interesting, and rather dramatic increase in the width of the peak of the first run at 123mA. We can see in figure 4.16 that this increase in width is apparent in every run of the data, and in fact peaks frequently appear around the transitions between the two regimes.

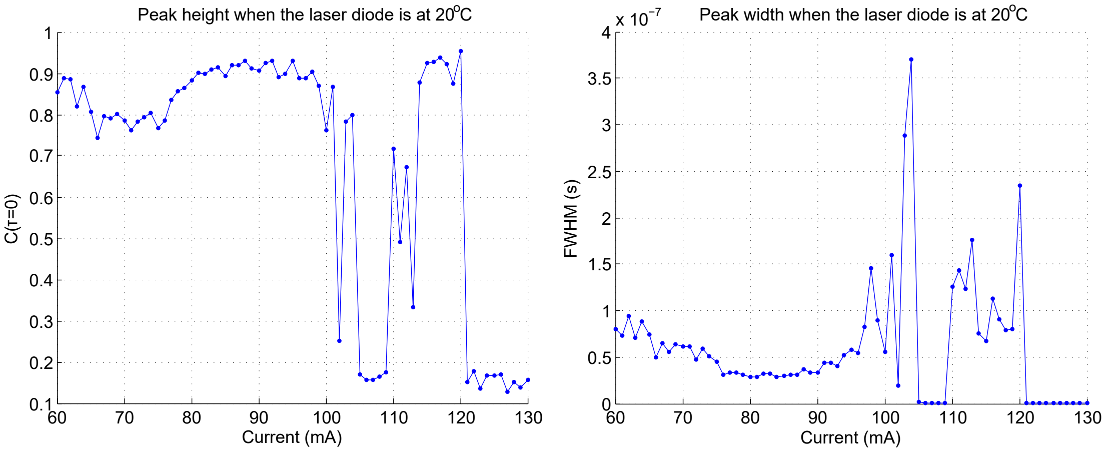
Figure 4.17 shows some of the individual traces from these higher peaks, and you can see that these are caused by the raw data showing transitions between the two regimes while the source to the laser remains constant. This transition between the two would explain the shape of the correlation, as the small fluctua-



(a) Repeats without turning the laser off between.

(b) Repeats where the laser is turned off between each.

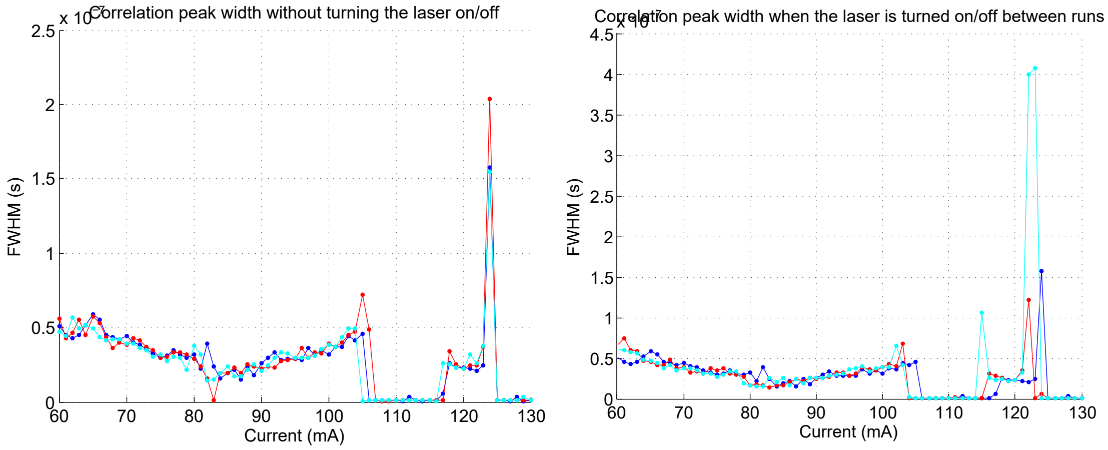
Figure 4.14: Variation in the location of transitions in the statistics of the light. Both blue traces on each plot were taken when increasing the current each step, and the red ones were taken running in the opposite direction, decreasing the current each step.



(a) Correlation heights at 20^{o}C.

(b) Correlation widths at 20^{o}C.

Figure 4.15: Correlation heights and widths at a gain chip temperature of 20^{o}C show a much messier transition region than at the usual 25^{o}C.



(a) Repeats without turning the laser off between.

(b) Repeats where the laser is turned off between each.

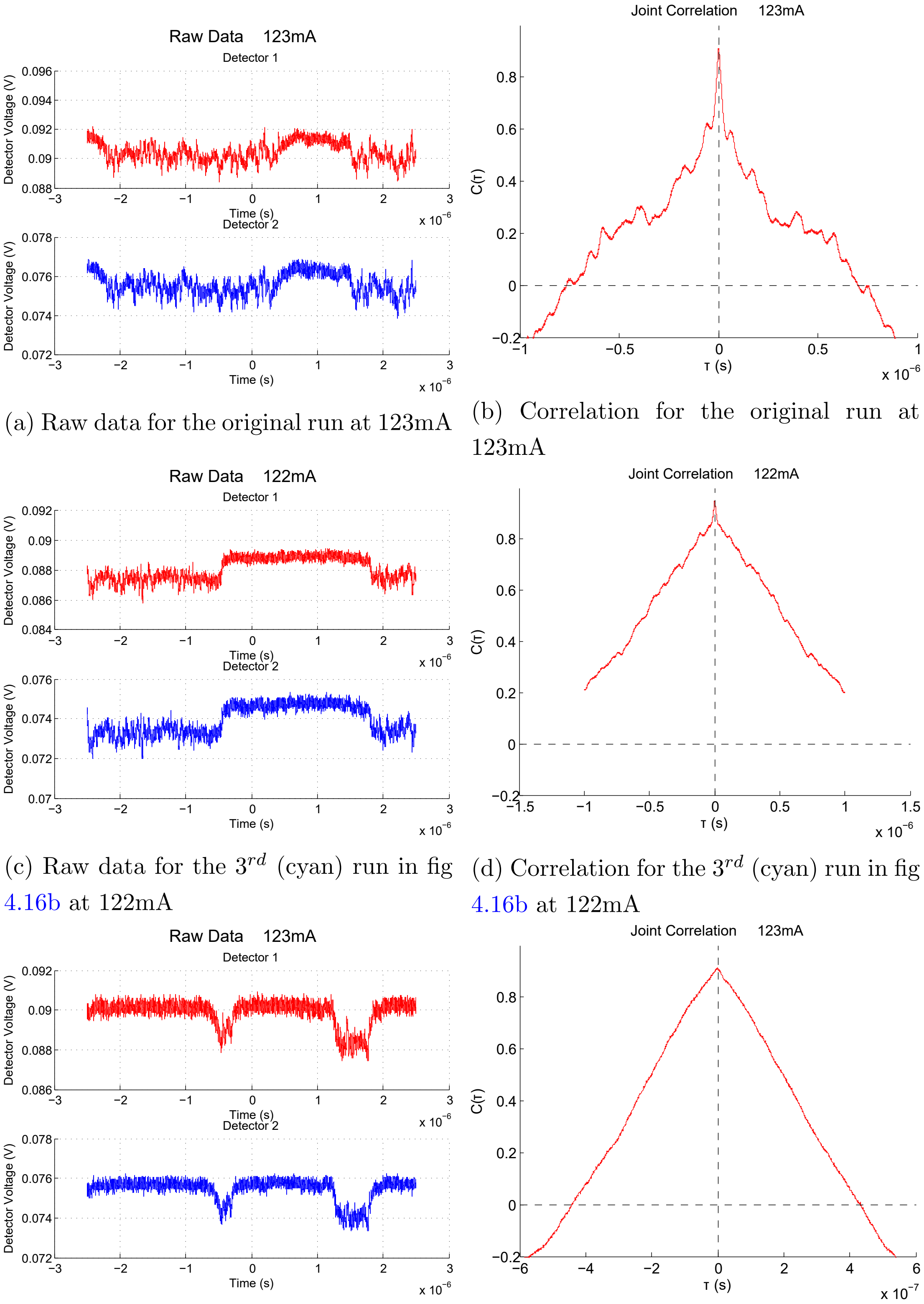
Figure 4.16: Variation in the location of transitions in the statistics of the light. Both blue traces on each plot were taken when increasing the current each step, and the red ones were taken running in the opposite direction, decreasing the current each step.

tions are drowned by the larger change in voltage associated with the transition.

4.2.5 Ensuring correct alignment

When setting up the experiment, it was noticed that the values of the correlation peaks shifted somewhat over long time-scales, on the scale of months or more (figure 4.18). Not only did the height of the peak change, but the position also strayed from $\tau = 0$. After some experimentation with the alignment of the system, it was found that this was due purely to the optical isolator. The better aligned the optical isolator, the higher the fluctuations, and the higher the peak.

To investigate this further, we took the optical isolator out entirely and then proceeded to re-run the experiment. It was found (figure 4.19) that this eliminated most of the correlations, and in some cases anti-correlations appeared. Where correlations were present, frequently the peak of the correlation had shifted significantly away from $\tau = 0$.



(a) Raw data for the original run at 123mA

(b) Correlation for the original run at 123mA

(c) Raw data for the 3^{rd} (cyan) run in fig 4.16b at 122mA

(d) Correlation for the 3^{rd} (cyan) run in fig 4.16b at 122mA

(e) Raw data for the 3^{rd} (cyan) run in fig 4.16b at 123mA

(f) Correlation for the 3^{rd} (cyan) run in fig 4.16b at 123mA

Figure 4.17: Raw and correlation(τ) data for the cases where the FWHM looked particularly high.

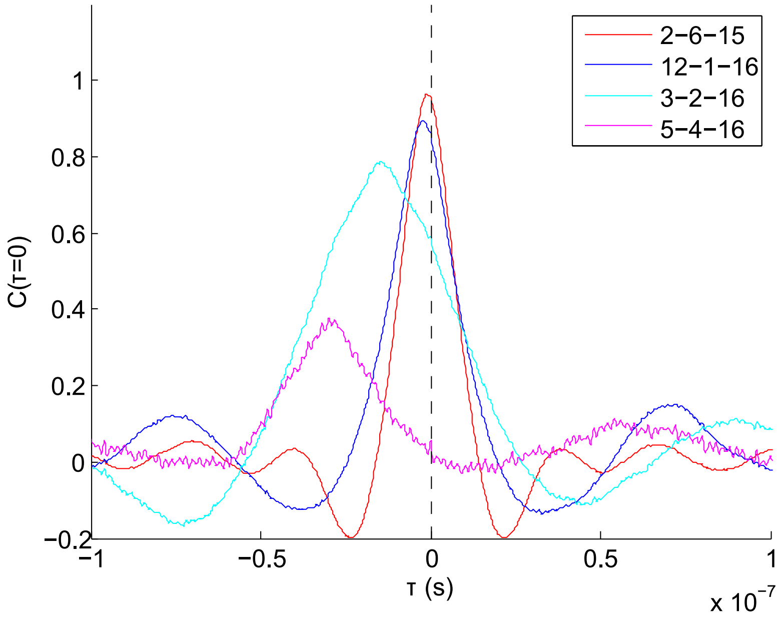


Figure 4.18: The drift in correlation peak height and position over time. The dates on which each trace was taken are marked in the legend.

This is due to reflections of the light which bounce off the optical components in the experiment, and re-enter the laser cavity, and then the gain chip. This reflected light will cause an unexpected increase in photon number in the cavity, thus triggering the production of more photons through stimulated emission.
A much smaller sample of data than usual was taken in this case, as we wanted to minimise any damage to the laser that may occur as a result of reflections from the equipment. If this was not a concern, it would have been interesting to extend the dataset to higher currents, as the effects seem to increase with current, and also to see if there was any effect in the lasing regime.

4.2.6 Fourier transform

The Fourier transform tells us which frequencies appear in our signal. Comparing the Fourier transforms for each detector tells us if one of these is subject to noise

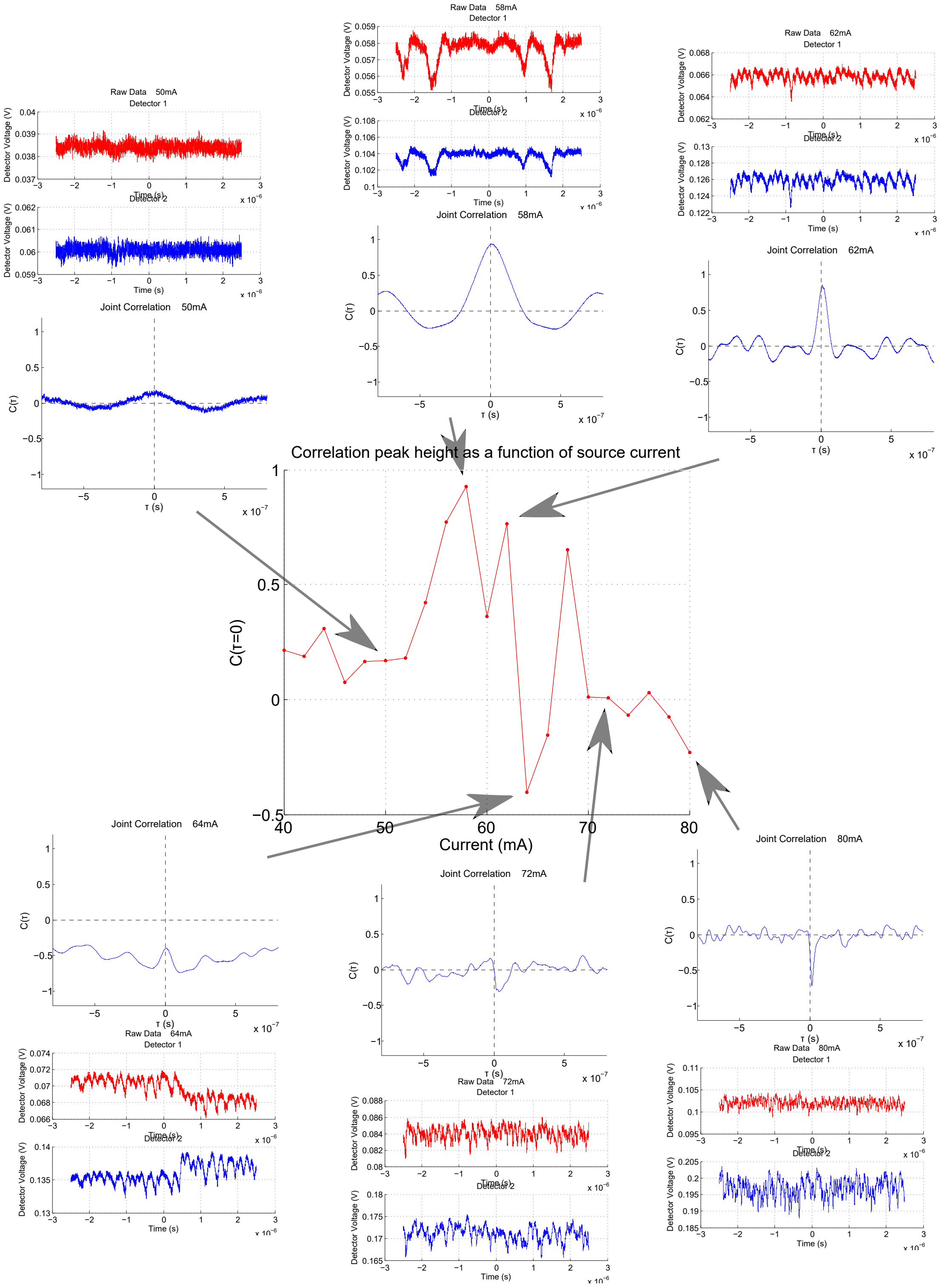


Figure 4.19: The effect on the correlations of removing the optical isolator from the experiment.

which the other isn't, and comparing transforms between the two regions tells us if there are any frequency components more prominent in one region than the other.
To achieve this, the output from the detectors was first normalised by subtracting the mean, and then multiplied by Chebyshev window function. Window functions are generally zero outside a specified range. As we are using a fast Fourier transform (FFT), where only a few frequencies are scanned, the windowing is necessary because there may not be a complete number of cycles from each frequency in the data set. We then use the MATLAB function fftshift to reorder the output of the FFT function so that the zero frequency data is in the centre.

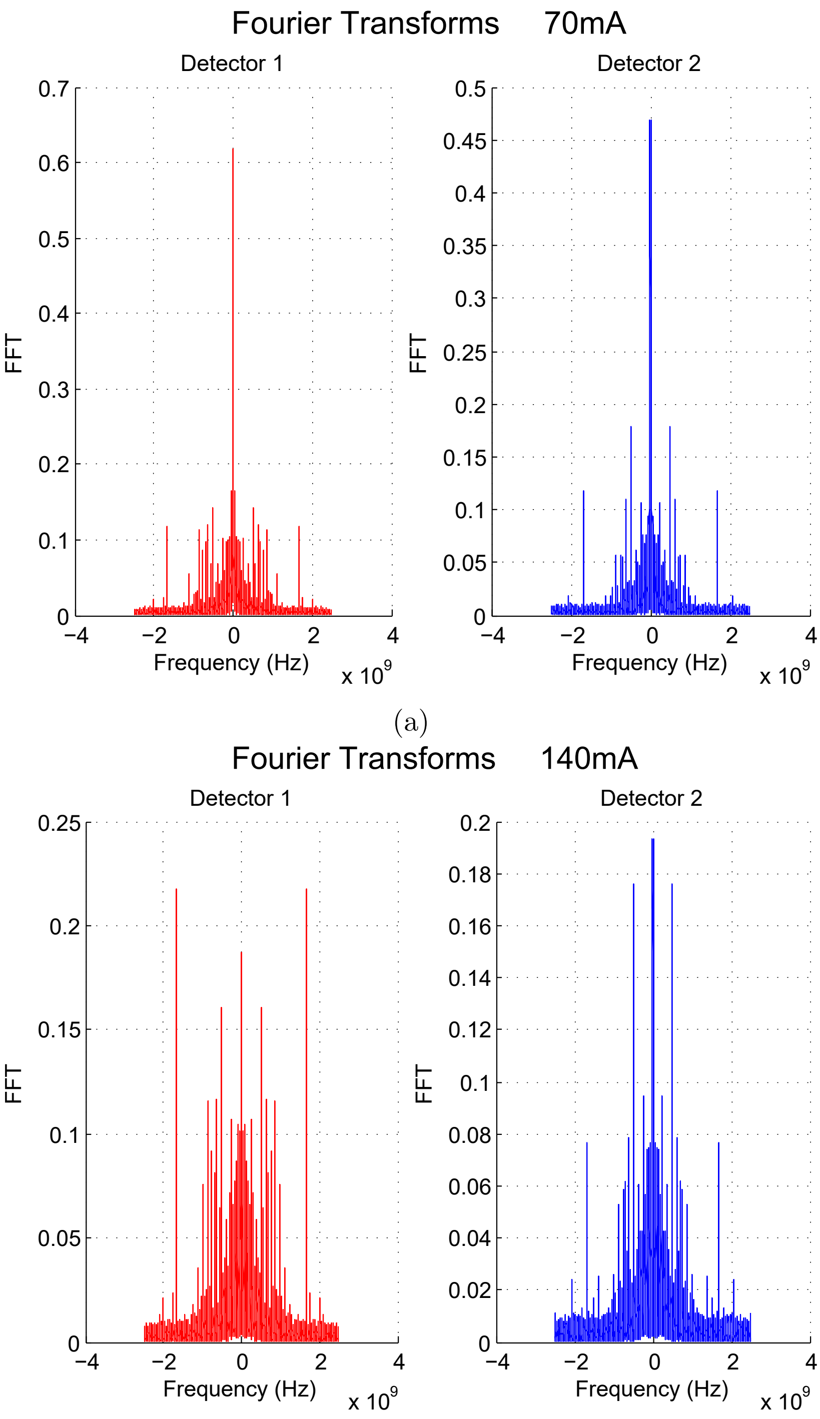
In figure 4.20, it becomes apparent that this data is quite difficult to interpret. Instead, a more useful measure of the frequency components of the signal is the power spectral density.

4.2.7 Power spectral density

The power spectral density shows us how the power in the signal is distributed as a function of frequency.

In figure 4.21, it is clear that there is very little real difference between the two detectors, although detector one continuously picks up slightly higher powers at the lower frequencies. This effect remains even if the positions of the detectors are exchanged. There is also no obvious difference between the two source currents, except for slightly higher power readings at the higher current, as you would expect.

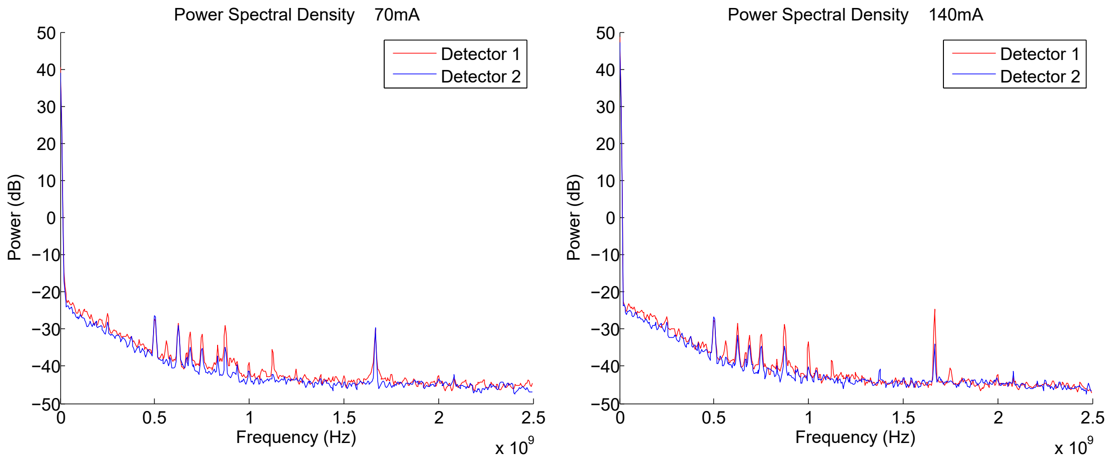
This means that there is no obvious external influences affecting one detector more than the other, or that affect the ASE regime in a different way to the lasing regime, or vice versa.



(a)

(b)

Figure 4.20: Fourier transforms of the experimental data at 70mA and 140mA.



(a)

(b)

Figure 4.21: Power spectral density of the experimental data at 70mA and 140mA.

4.2.8 Ruling out noise

The intensity of the light leaving the source is governed by the size of the current through the gain chip. A higher current means more light. We need to check that the intensity fluctuations we see at low operating currents are not due to fluctuations in this current.

We tried a number of ways to see if the source current noise was having an effect. First, we hooked up the "Analogue control out" output from the back of the laser control module to the oscilloscope. This output give a value of between 0V and 10V, proportional to the actual current output of the control module, via the equation

$$U_{out} = V_0 \frac{I_{act}}{I_{max}}, \tag{4.2}$$

Where U_{out} is the voltage output from the monitoring jack, V_0 is 10 V, I_{act} the actual current to the gain chip, and $I_{max} = 1A$, the maximum possible output current[1].

[1] In the ITC001 manual text, the output jack is referred to as R1 instead of R6.

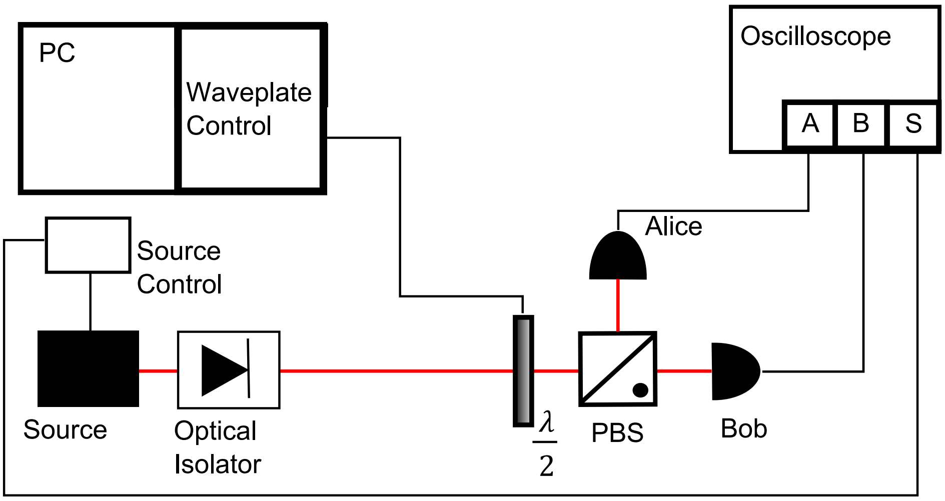


Figure 4.22: Experimental set-up to monitor the source current fluctuations.

By recording this data along with the detector output data in the oscilloscope, (figure 4.22) we could then see if there were any correlation between the current monitoring output and the intensities recorded by the detectors. This is shown in figure 4.23.

The blue line is the correlation between the two detectors, Alice and Bob. This is off centre slightly, and significantly below the usual correlation values due to the optical isolator having drifted slightly away from perfect alignment (section 4.2.5). While this isn't optimal, it is still well enough aligned not to cause any issues other than the strength and centring of the correlation peak.

The pink and red lines show the correlation between the current monitor source control output and each of the detectors. If there were fluctuations in the control current which influenced the laser output, you would expect to see peaks in both of these lines. These are absent, suggesting that fluctuations in intensity in the output of the laser are not influenced by fluctuations in the source current.

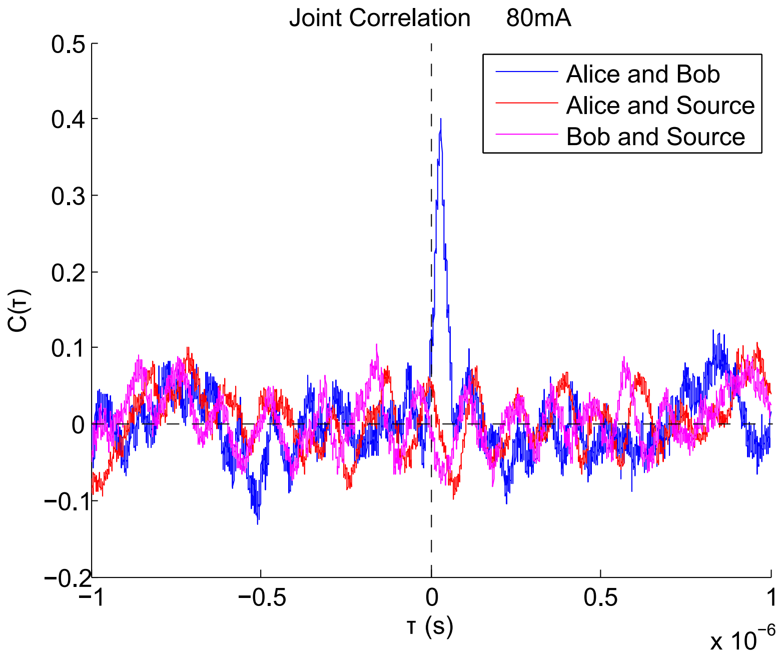


Figure 4.23: Correlations between detectors and source current fluctuations.

4.2.9 The effect of loss

The purpose of this section was to rule out the possibility of the detectors being the source of the correlations. We wanted to find out whether it was possible that at low intensities, the higher proportion of detector noise could cause fluctuations of the data. This task was made a bit easier as we only needed to rule out noise source which could cause correlated fluctuations in the outputs of both, separate, detectors. For example, most electrical noise could be immediately discounted, as the detectors are battery operated, so do not share a common power supply. The most obvious remaining source of noise is any stray background light entering the detectors.

There are a number of ways to test for this, for example, one being to add a third detector into the system, but not shine any of the light beam into it, then see if that correlates with the two other detectors. This however would be tricky, as there is not a lot of background light. In the end, we decided that the most interesting would be as follows.

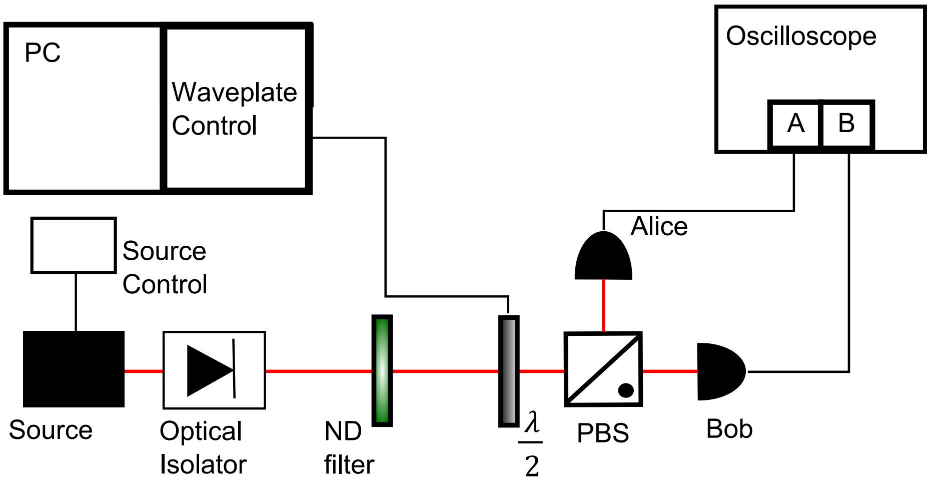
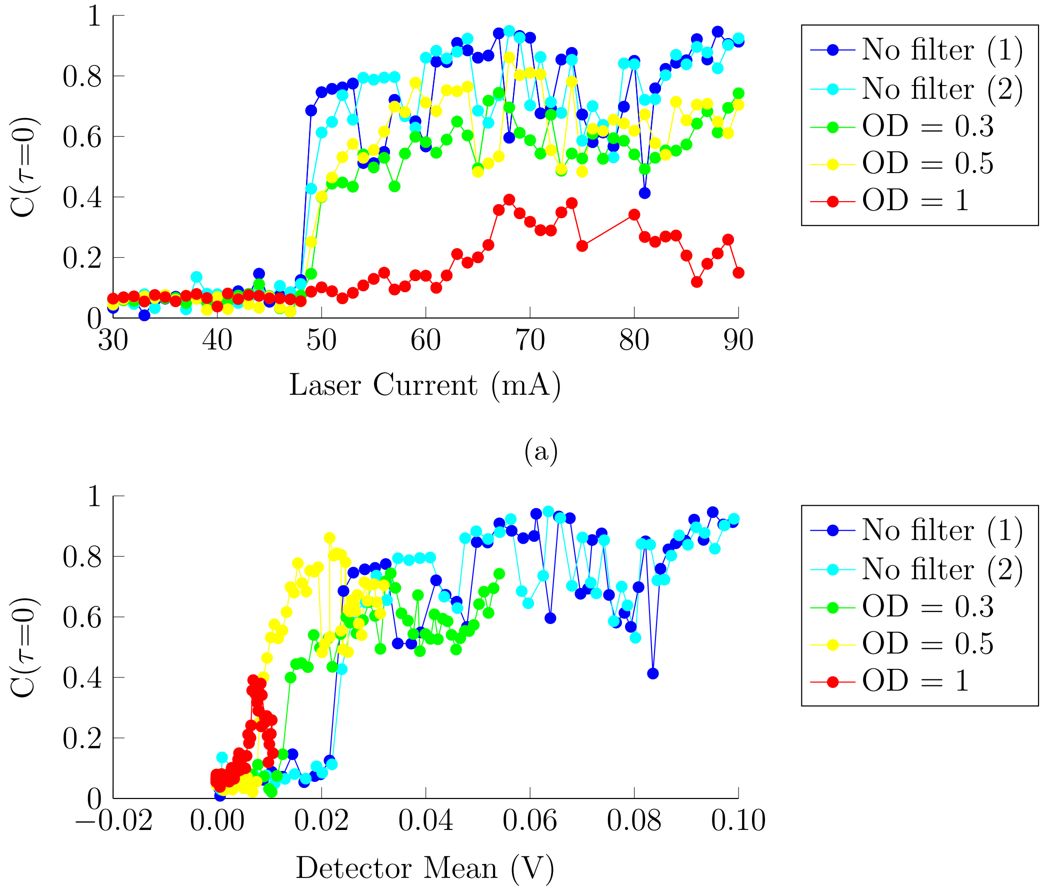


Figure 4.24: Experimental set-up to examine the result of loss. ND filters of various strengths were inserted into the beam path.

The detectors only registered large fluctuations, and therefore correlations, when the laser current, and thus the intensity of the light, were low. If the fluctuations at low current levels were due to the intensity of light being too low when it reaches the detectors, then artificially lowering the intensity of light produced at high source currents should also produce fluctuations in detection.

Artificially lowering the intensity of the light is easy. We placed neutral density (ND) filters of various strengths into the path of the beam before the beamsplitter, as in figure 4.24. ND filters are rated with an optical density (OD). The higher the value of the optical density, the more light is lost.



(a)

(b)

Figure 4.25: Correlation peaks as (a) a function of source current, and (b) detector voltage, for various strengths of ND filter. The grouping of transitions in the current graph indicates that it is the current rather than the intensity which causes this transition.

We then ran the experiment as before, measuring the height of the correlation peaks, and plotting them against the laser current (4.25a) and the mean detector voltage (4.25b). This last was used as a measure of how bright the light from the source was when it reached the detector. To keep the measurements consistent, the laser was not turned off between runs. There is a clear grouping of transitions into the ASE regime in the current graph, whereas in the intensity graph, no grouping is present. This suggests that, as hoped, the source current is the reason for the fluctuations, rather than any affect of low intensity or any background noise on the lasers.

It is interesting to note however, that in the presence of increased loss, where the optical density of the ND filters was higher, the magnitude of the correlations was smaller. As this work is primarily about cryptography, especially ways in which someone syphoning off a portion of your signal can be detected, this is worth keeping in mind. Unfortunately, in this small test-case, it looks like there is not a particularly strong relationship between the amount of loss and the reduction in correlation, but it would be worth exploring further.

4.2.10 Creating pseudo-thermal light

In sources such as [113–115], a method of inducing fluctuations into a coherent beam to create pseudothermal light is shown. A ground glass disk is placed into a laser beam, and rotated. The ground glass disk produces a speckle pattern of various intensities by the superposition of the diffracted contributions of the light through the glass. As the disk is rotated, the speckles move, creating fluctuations in the intensity of the light, mimicking those produced by a true thermal source.

As the kit laser we use is no longer in production, we began examining the possibility of using a similar set-up in future.

Instead of transparent ground glass disks, we used diffuse reflectors (ThorLabs DG10), in a variety of different grit polishes. As these produce a large amount of diffusion, lenses were needed to attempt to collimate the light sufficiently for the intensity at the detectors to be high enough. The diffuse reflector was mounted onto the same type of rotating mount as the half wave plate, and controlled in the same way, (see section 4.2.4). The set-up is shown in figure 4.26.

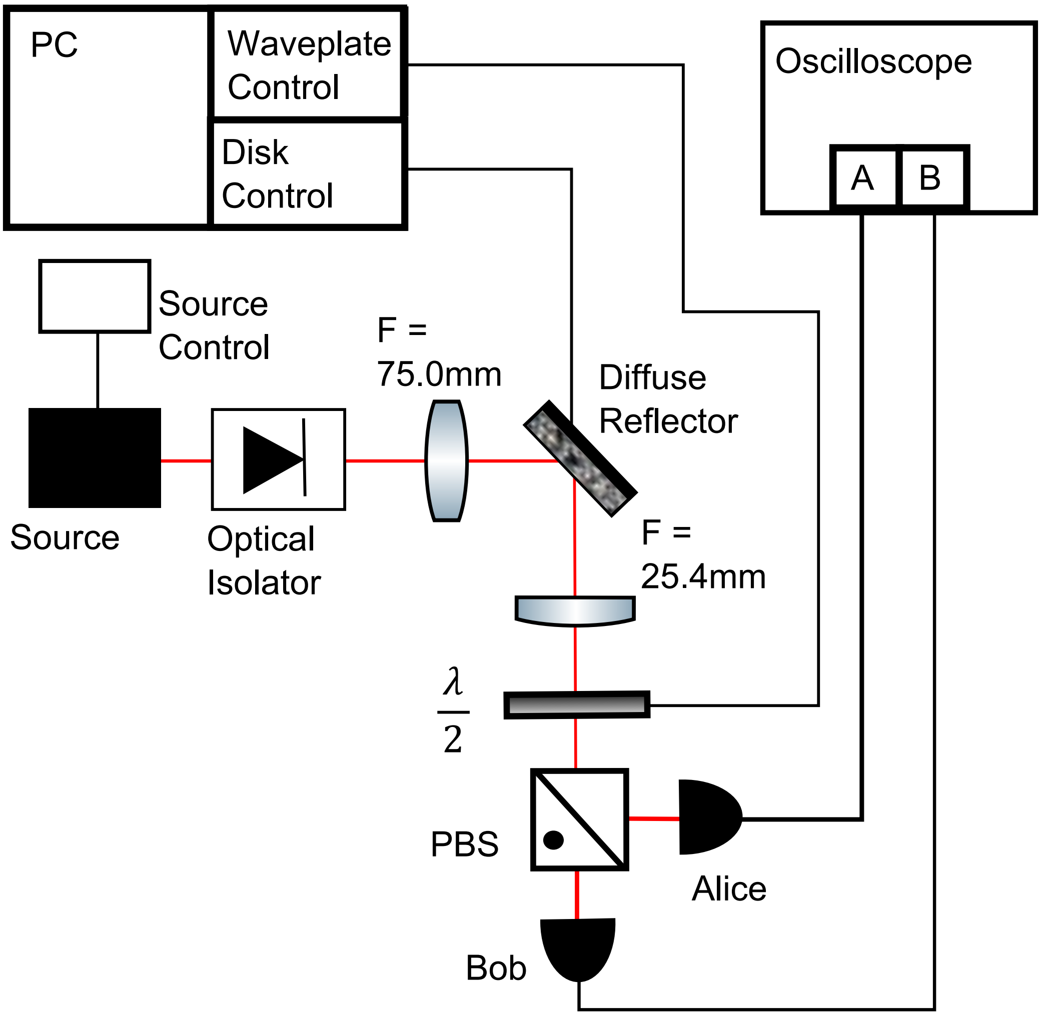


Figure 4.26: Experimental set-up to produce pseudothermal light.

The source was run at the recommended lasing operating current of 140mA, well away from the ASE regime, so no fluctuations from the source would interfere. This was shone through the optical isolator, then the beam narrowed via a biconvex lens with a focal length of 75.0mm. This was a balancing act of trying to narrow the beam sufficiently to try and limit the diffusion off the reflector, while also covering sufficient area of the diffuser to produce speckles. From here, the beam was reflected off the diffuser, and through a plano-convex lens with a focal length of 25.4mm in order to try and collimate the diffuse light. This was then passed through the beamsplitting arrangement (as in section 4.2.4) and to the detectors. The output of which was recorded on the oscilloscope. The spin of the reflector was governed by the ThorLabs APT control software.

Unfortunately, due to the diffuseness of the light from the reflector, alignment was particularly challenging. It was also noticeable, that no clear speckle pattern was observed by eye when a piece of IR card was inserted directly after the reflector. It is likely that even the coarsest reflector created light that was too evenly diffuse to create pseudothermal light. This is after all not the primary purpose of these reflectors.

It was found during experiments, that no correlations existed with the diffuser in place. This includes when it was spinning, and when stationary. Only one speed was avaliable on the spinner, although the speed of the spinner should only determine the coherence time of the pseudothermal beam. When the disk is stationary, there should be no correlations because the fluctuations are over space rather than time. Data was taken over a time period of 5μs, meaning that for there to be a noticeable number of fluctuations in that time, at least 5 say, then the disk would need to be rotating to pass a million speckles per second. The diffusers we used ranged from 120grit (lumps 0.01 ± 0.02mm) to 1500grit (lumps $1.0 \pm 0.2\mu$m). The disk rotated at 25^o/s, and had a radius of 12.7mm. We know that $\frac{speckles}{second} = \frac{speckles}{meter} \times \frac{meters}{degree} \times \frac{degrees}{second}$. We have an target $\frac{speckles}{second}$, we know $\frac{degrees}{second}$, and we have a range of values for both $\frac{speckles}{meter}$ and $\frac{meters}{degree}$ due to whether the laser spot is focused towards the centre or the edge of the disk. A spot towards the centre of the disk ($r_{inner} = 2$mm) would travel at 0.3mm/s, whereas one towards the edge ($r_{outer} = 10$mm) would do 1.4mm/s. This means that we can achieve speckles/second in the range 30-1400. Several orders of magnitude too slow[1].

A very easy fix for this would be to collect data over a period of 5ms rather than 5μs. The oscilloscope is perfectly capable of doing this although it does take a while to process that much data. A significant amount of effort would also be needed in order to try and align the components such that a usable intensity of light reaches the detectors. Unfortunately, this project was shelved before this could be completed.

If the project had been continued, it would have provided a valuable resource. Our current set-up had a very small height of the $g^{(}2)$ peak, due mainly to the short coherence time of the light source. Using a laser and ground glass disk to create pseudothermal light in this way allows us to choose a much longer coherence time. This would make the physical alignment of the subsequent experiment much easier, as well as increasing the height of the $g^{(}2)$ peak.

[1] If there is one thing I have learnt from this PhD it is to insist on doing these sorts of calculations *before* getting tied up in the experiment.

Chapter 5

Secrecy in Thermal States

5.1 Experimental demonstration of a central broadcast scheme

To begin examining this scenario, of using a central broadcast scheme with continuous variables, a rough experiment was created with the aim of judging the feasibility of such a system. In addition, the possible use of thermal states as a means of increasing the secrecy was examined.

5.1.1 Experimental Methods

The experimental set-up was fairly simple, as this was only a proof of principle experiment. The light source, which could be switched between thermal and coherent output, was passed through two variable beam splitters so that the proportion of the beam arriving at each of three detectors could be controlled. This allowed us to simulate different levels of noise. The set-up is described in figure 5.1. While it could be argued that this is not *strictly* central broadcast, experiments are currently under way by Guiazon to translate this experiment into the microwave regime.

The TLK - 780M laser (the same as used in Chapter 4) was first passed through an optical isolator, to protect the laser components from any stray re-

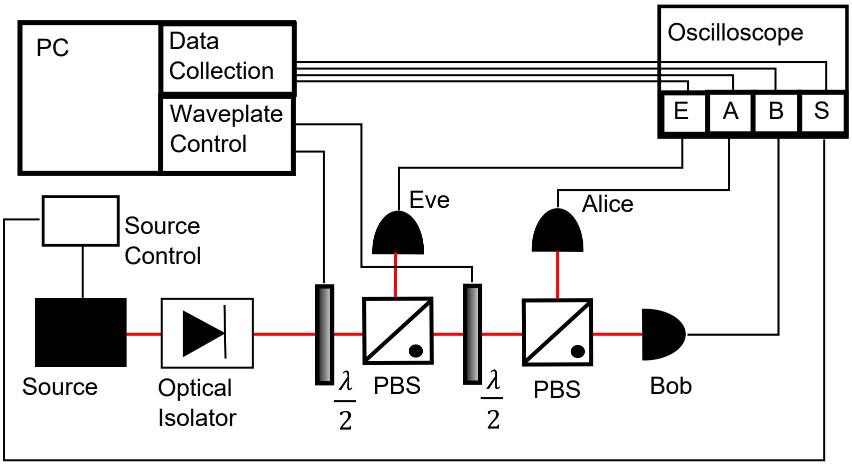


Figure 5.1: Experimental set-up for the proof of principle central broadcast protocol. The combination of half wave plates and polarising beam splitters act as variable beam splitters, allowing us to control the proportions of light reaching each detector and thus simulate different noise levels.

flections. It also has the added advantage of ensuring the transmitted light is linearly polarised. This light was then passed through a combination of a half wave plate set in a computer controlled rotating mount, and a polarising beam splitter. The combination of these two components acts as a variable beam splitter. The half wave plate rotates the polarisation of the light, while the polarising beam splitter sends the horizontally polarised component towards the first detector (Eve), and the vertically polarised component onwards. This is then followed by a second variable beamsplitter identical to the first. This time however, the light is split only between the two remaining detectors. As in the previous chapter (chapter 4), the outputs of the detectors are connected via BNC cables to the LeCroy Waverunner 44xi oscilloscope. However this time, the oscilloscope was controlled via a python script on the PC to collect the data and save it to file. This python script also controlled the rotation of the half wave plates, and thus the intensity of light reaching each of the three detectors. The beauty of this system is that you don't have to re-run the experiment if you want to move the relative positions of Alice Bob and Eve, you can just re-label the detectors, and re-use the same data.

To avoid confusion during the re-labelling process, each of the detectors and spinners was numbered, as in figure 5.2.

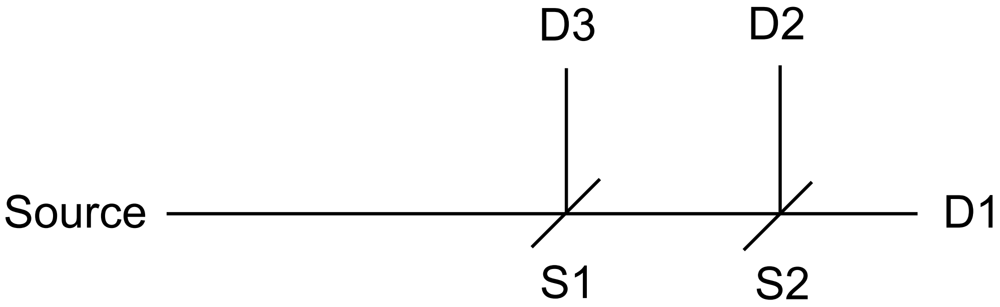
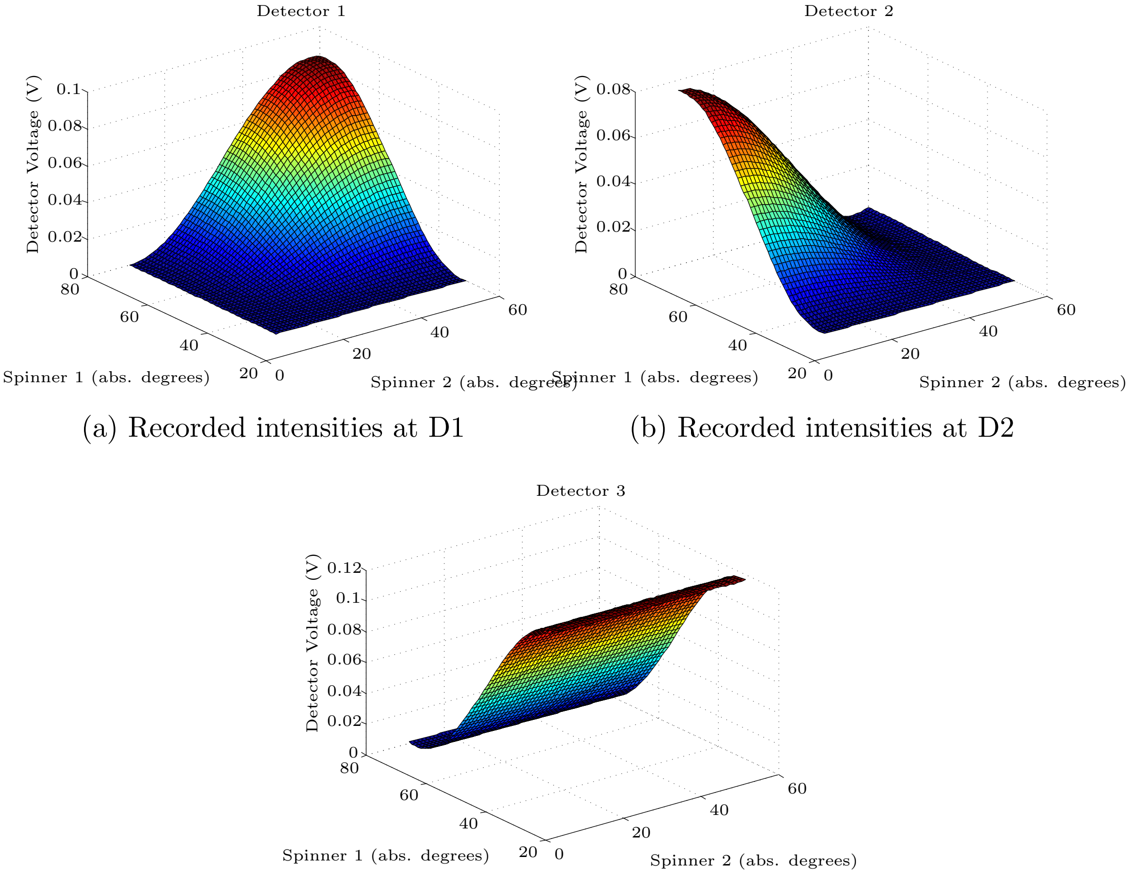


Figure 5.2: A simplified schematic of the central broadcast set-up showing the numbering used for each detector and half wave plate.

5.1.2 Validation

The oscilloscope records a set of continuous data from each detector. The python script saves all this data to file, where it is then transferred via USB drive to another computer where the analysis is carried out in MATLAB.

The first stage in this process is to check that as the spinners rotate, the intensity at the detectors changes. This is shown in figure 5.3. As the ND filters are not marked with a zero position, the starting position in degrees is also non zero, and different between the two spinners. Each spinner was spun over a range of 49^o to allow a little overlap at the edges, and ensure that the join was smooth. Strictly, only 45^o should be needed as a half wave plate rotates the polarisation direction of the light by 2θ, where θ is the angle between the polarisation direction of the incident light, and the fast axis of the wave plate. This is explained in more detail in section 4.2.1. The 45^o rotation of the half wave plate produces a 90^o rotation of the polarisation of the light on the other side of it, which is



(a) Recorded intensities at D1

(b) Recorded intensities at D2

(c) Recorded intensities at D3

Figure 5.3: Detector intensity values for each of the three detectors as a function of spinner rotation for the two half wave plates.

sufficient to go from being completely horizontally polarised, where all the light passes through the polarising beamsplitter in one direction, to being completely vertically polarised, where all the light passes through the beamsplitter in the other direction. Care was taken when choosing the start points of the rotators that both these extremes were covered during the 49^o of rotation.

Next, it is important to establish that the correlations in the thermal light can still be detected when the detectors have different path lengths from the various beamsplitters. In section 4.2.4, it was established that to observe the highest correlations, the detector path length should differ by no more than 1.5mm, but correlations were still observable when this condition was not met. Ideally, here, all three detectors should have the same path length, but again, in reality this has proved tricky to implement. To check that correlations can exist between the detectors, points were chosen where the intensity of the detectors in question was high. This is because, it was shown in section 4.2.9 that high loss could reduce the peak height of the correlations. Figure 5.4 shows the $g^{(2)}$ correlations between each pair of detectors. The magnitude of the peak height here is smaller than in section 4.2, this is mostly due to a combination of lower detector intensity, due to the addition of the third detector, and the additional splitting of the light this involves. However, the differences in peak height between the three pairs of detectors is due to the differences in path lengths.

For comparison, we also present graphs of the $g^{(2)}(\tau)$ for an area where one detector receives a high intensity of light, and the other a low intensity (figure 5.5a). No peak in the $g^{(2)}$ is discernible in this case, but the trace looks cleaner than when the source is thermal rather than coherent (figure 5.5b) and the spinner settings are the same as in figure 5.4a.

5.1.3 Key rate

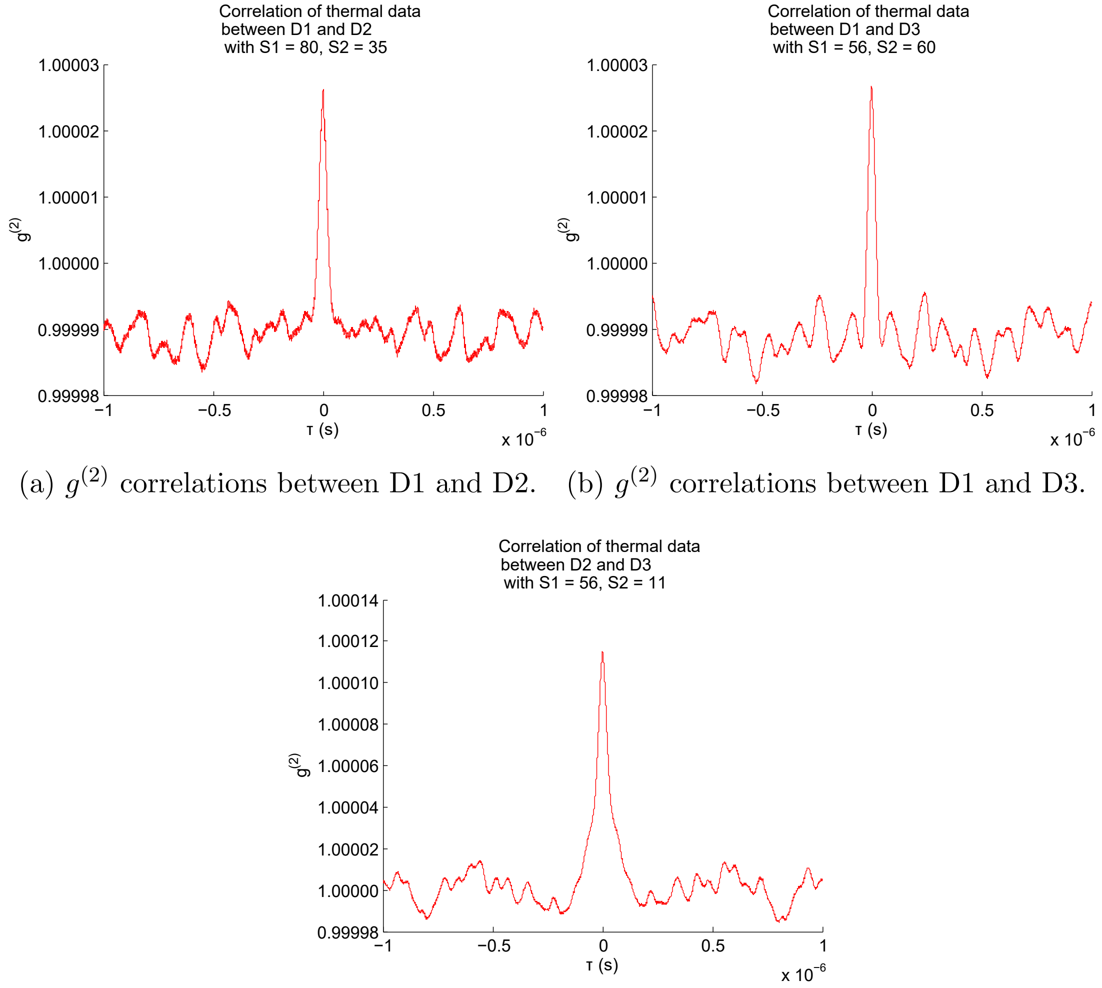
In QKD protocols, the level of mutual information between Alice and Bob is not a reliable indicator of the success of the protocol, as this omits any reference to the eavesdropper. Instead, a measure called the secret key rate is usually used. As with all things in QKD, several definitions are used.

In [81, 116], Maurer defines the secret key rate as

$$\mathrm{S(A;B||E)} \geq \max[\, \mathrm{I(A;B)} - \mathrm{I(A;E)},\ \mathrm{I(A;B)} - \mathrm{I(B;E)}\,] \tag{5.1}$$

where A, B and E are the datasets after the quantum transmission of Alice, Bob and Eve respectively. These quantities are illustrated in figure 5.6.

In [116], he also described a property called the "intrinsic conditional mutual information", $\mathrm{I(A;B \downarrow E)}$, which acts as an upper bound on the secret key

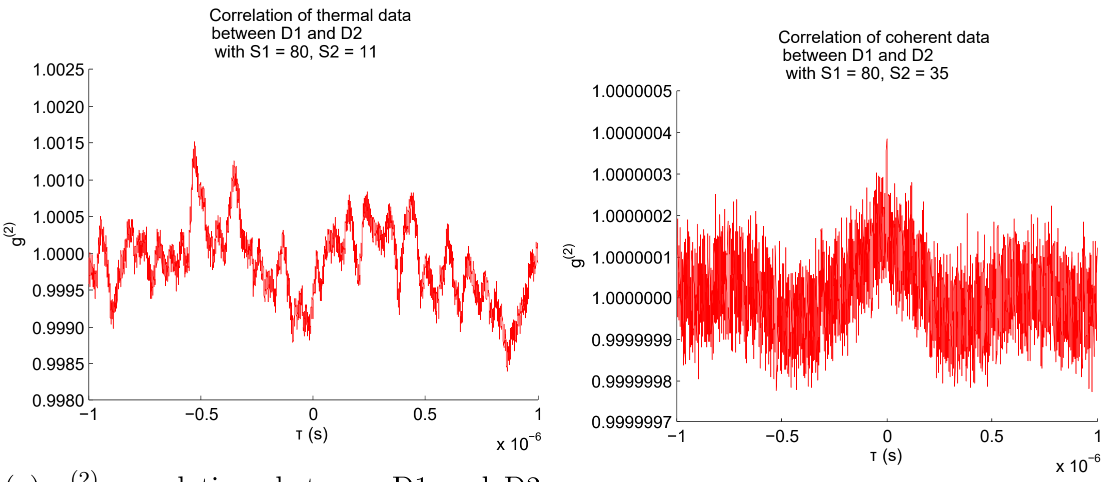


(a) $g^{(2)}$ correlations between D1 and D2.

(b) $g^{(2)}$ correlations between D1 and D3.

(c) $g^{(2)}$ correlations between D2 and D3.

Figure 5.4: $g^{(2)}$ correlations in the central broadcast set-up using thermal light. These were taken in regions where the detector intensities of each pair were approximately equal.



(a) $g^{(2)}$ correlations between D1 and D2 where D1 receives the minimum amount of light, and D2 the maximum.

(b) $g^{(2)}$ correlations between D1 and D2 in the coherent regime.

Figure 5.5: Examples of the absence of $g^{(2)}$ correlations for areas with vastly differing intensities, or in the coherent regime.

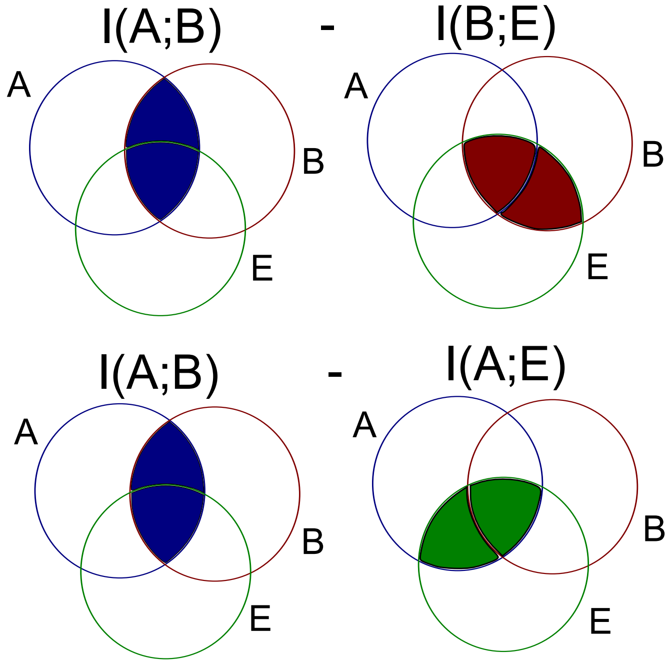


Figure 5.6: A Venn diagram showing $\mathrm{I(A;B)} - \mathrm{I(A;E)}$ and $\mathrm{I(A;B)} - \mathrm{I(B;E)}$

rate. This is done by sending Eve's information over an arbitrary channel which maximises I(A; B; E) (Sometimes also referred to as R). This is shown in figure 5.7.

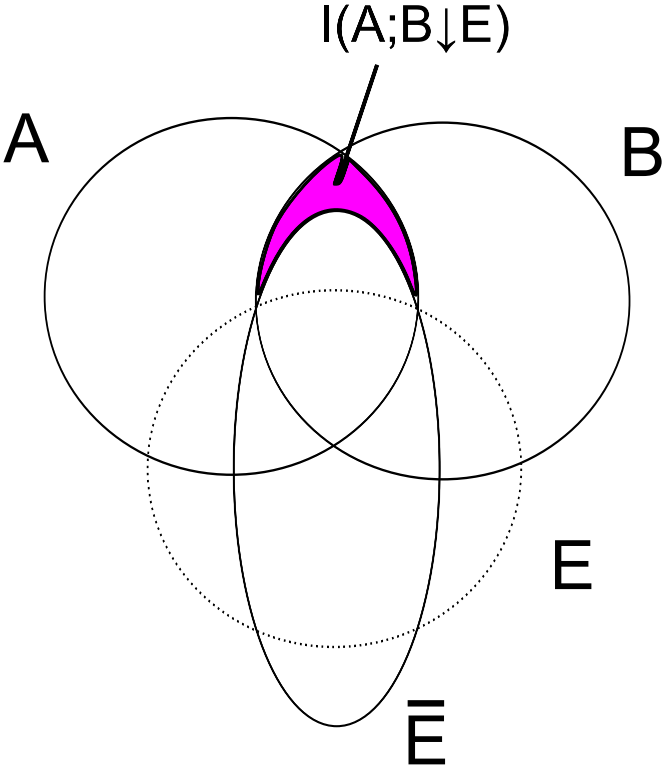


Figure 5.7: A diagram showing the intrinsic conditional mutual information, I(A; B ↓ E), based on that in [116]. $\bar{E}$ is the maximum amount of information Eve can gain about I(A;B) by sending her received data over an arbitrary channel. This channel in effect distorts her information into the shape shown above. She now knows less about both H(A) and H(B) but more of I(A;B).

Additionally, in [79], Grosshans and Grangier introduce a technique called "reverse reconciliation", where they decide that they can beat a 3dB loss limit associated with CVQKD, by causing Alice to change her data to match Bob's rather than the usual Bob changing his data to match Alice's. This idea has recently fallen slightly out of favour. They use a secret bit rate of $\Delta\mathrm{I}_{\mathrm{RR}} = \mathrm{I}(\mathrm{A};\mathrm{B}) - \mathrm{I}(\mathrm{B};\mathrm{E})$ rather than the usually used $\Delta\mathrm{I}_{\mathrm{DR}} = \mathrm{I}(\mathrm{A};\mathrm{B}) - \mathrm{I}(\mathrm{A};\mathrm{E})$.

For the purposes of simplicity, and continuity with the majority of the field, here we use the key rate defined as $\Delta \mathrm{I} = \mathrm{I}(\mathrm{A};\mathrm{B}) - \mathrm{I}(\mathrm{A};\mathrm{E})$, and the *secret* key rate as $\mathrm{S}(\mathrm{A};\mathrm{B}||\mathrm{E}) \geq \max[\,\mathrm{I}(\mathrm{A};\mathrm{B}) - \mathrm{I}(\mathrm{A};\mathrm{E}),\ \mathrm{I}(\mathrm{A};\mathrm{B}) - \mathrm{I}(\mathrm{B};\mathrm{E})\,]$

5.1.4 Party planning

In order to examine a number of different possibilities, we look at two different protocols, each with a different arrangement of the three parties.

Protocol 1 imitates a wire-tap channel, where Eve is positioned between Alice and Bob. In our terms, that means that D1 is Bob, D2 is Eve and D3 is Alice. This is demonstrated in Figure 5.8.

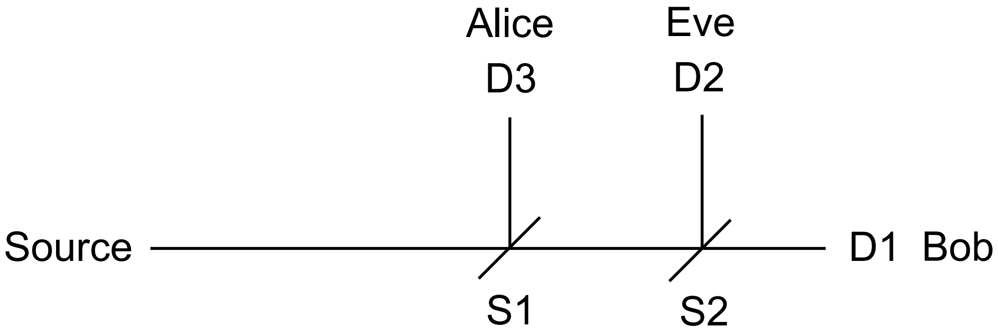


Figure 5.8: A simplified schematic of the central broadcast set-up for protocol 1 where Eve sits between Alice and Bob.

Protocol 2 is with Eve out the front so that she can receive most of the signal. As the amount of signal received by each party is controlled purely by the rotation of the half wave plates, the position of the relevant parties seems as if it should be irrelevant. We examine both to see if it makes any difference. Here, D1 is Bob, D2 is Alice, and D3 is Eve, as in figure 5.9.

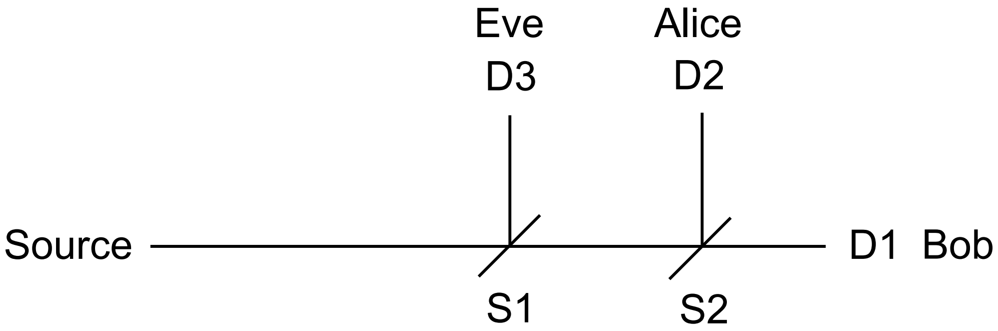


Figure 5.9: A simplified schematic of the central broadcast set-up for protocol 2 where Eve sits at the beginning of the experiment.

5.1.5 Calculating the mutual information

The main post processing begins by calculating the mutual information between each pairing of the three parties. Mutual information is the most basic ingredient needed for sharing a key. If there is no mutual information between Alice and Bob, there can be no key. The data here has not been modulated, as this is purely a feasibility study. This means that it would be incorrect to discretise, or divide the data into bits, before calculating the mutual information. This instead has to be done via the following equation.

$$I_{A,B} = \log\left(\frac{V_B}{V_{B|A}}\right) \tag{5.2}$$

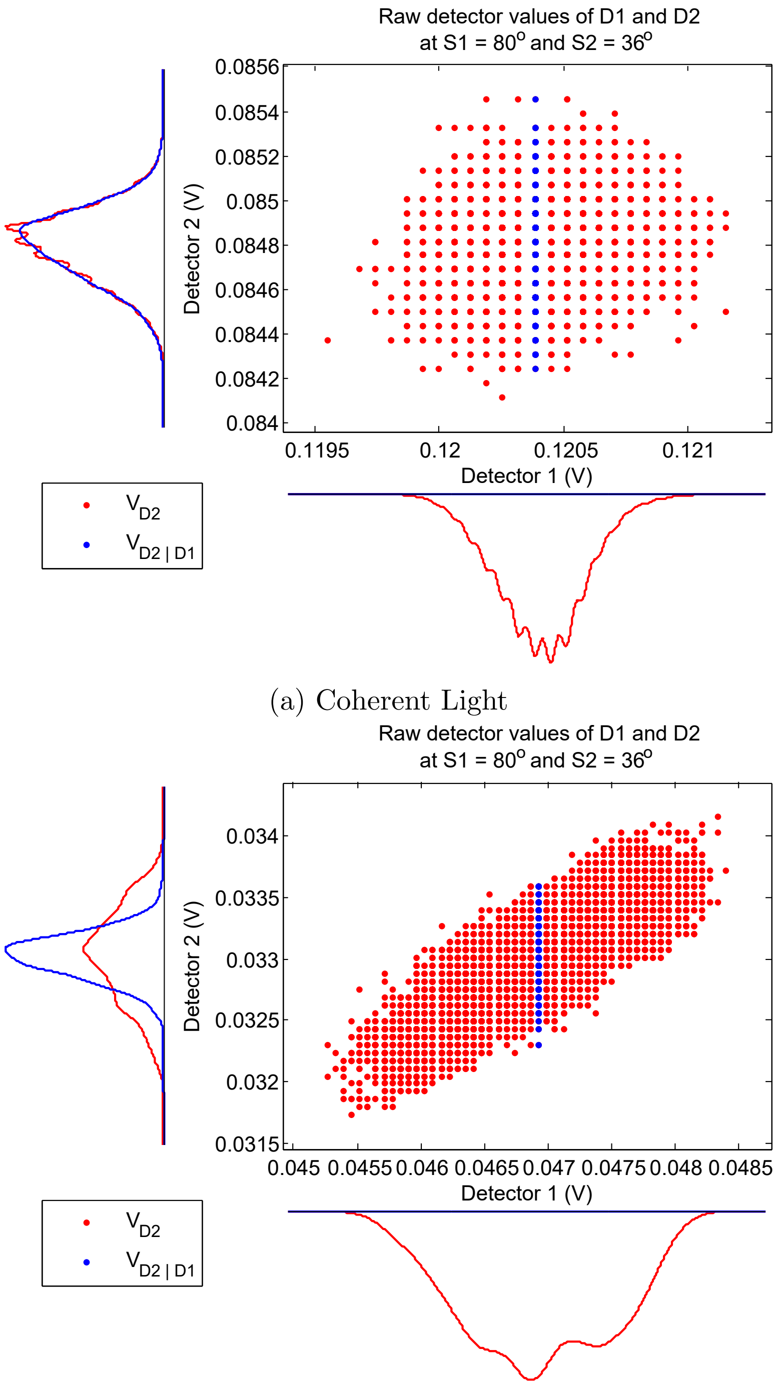
Where $I_{A,B}$ is the mutual information between two continuous datasets, A and B, V_B is the variance of dataset B, and $V_{B|A}$ is the variance of B at a given value of A. This is shown in figure 5.10, where the intensity each detector records is plotted in what is sometimes termed a “sausage graph”, due to the shape they tend to describe. These provide a very useful visual guide to the mutual information between two datasets. The longer and thinner the data sausage, the more mutual information between the two parties. The more spherical the sausage, the less mutual information. Equation 5.2 is simply a mathematical way of describing the sausage width. The shape arises due to the fluctuations in the light field. As the field fluctuates, both detectors record the change, so higher values in one detector will correlate with higher values in the second.

It is noticeable in figure 5.10 that the data points of both graphs exist on a grid pattern. This is due to the precision of the oscilloscope. The scope has a vertical resolution of 8 bits, and we were looking at very small fluctuations in the intensity of light, and thus the voltage produced by the photodiodes. The smallest vertical scale on the scope is 2mV/div. Over a range of, say, 10 divisions, that's 20 mV. $\frac{20mV}{2^8} = 0.08mV$ which rounds to the 0.1mV precision we see on the graphs. At this point, you might well say something like "That's a high end scope, why is the precision so rubbish?", and the answer to that, as with all things, is physics. The level of noise in 1Hz in a 50Ω system is -174dBm/Hz. The oscilloscope has a bandwidth of 150MHz, so the thermal noise at the input is $-174 + 10\log_{10}(150 \times 10^6) = -92$dBm. This is $10^{\frac{-92}{10}}$mW, or $V = \sqrt{WR} = \sqrt{6 \times 10^{13} \times 50} = 5\mu$V. It is estimated that the amplifiers in the scope will increase that number by a factor of 10, to 50μV. This means that out of our 1000μV precision, at least 50 of those are due to pure physics noise, which fundamentally *cannot* be reduced by things like better component design. If the resolution of the scope was increased to 12 bits, for example, then the precision would be 5μV, well inside the noise floor. So, in summary, there is no reason to increase the precision of the oscilloscope, as if you did so, noise would become an issue. Periodic larger gaps that may be visible in the thermal data (figure 5.10b) are an artefact of MATLAB's rendering process.

The mutual information between each pair of detectors across the whole range of spinner rotations are shown in figure 5.11.

In attempt to make these graphs easier to understand and a little more generalised, they have also been plotted against normalised detector intensity rather than spinner rotation. The relationship between these is shown in figure 5.12.

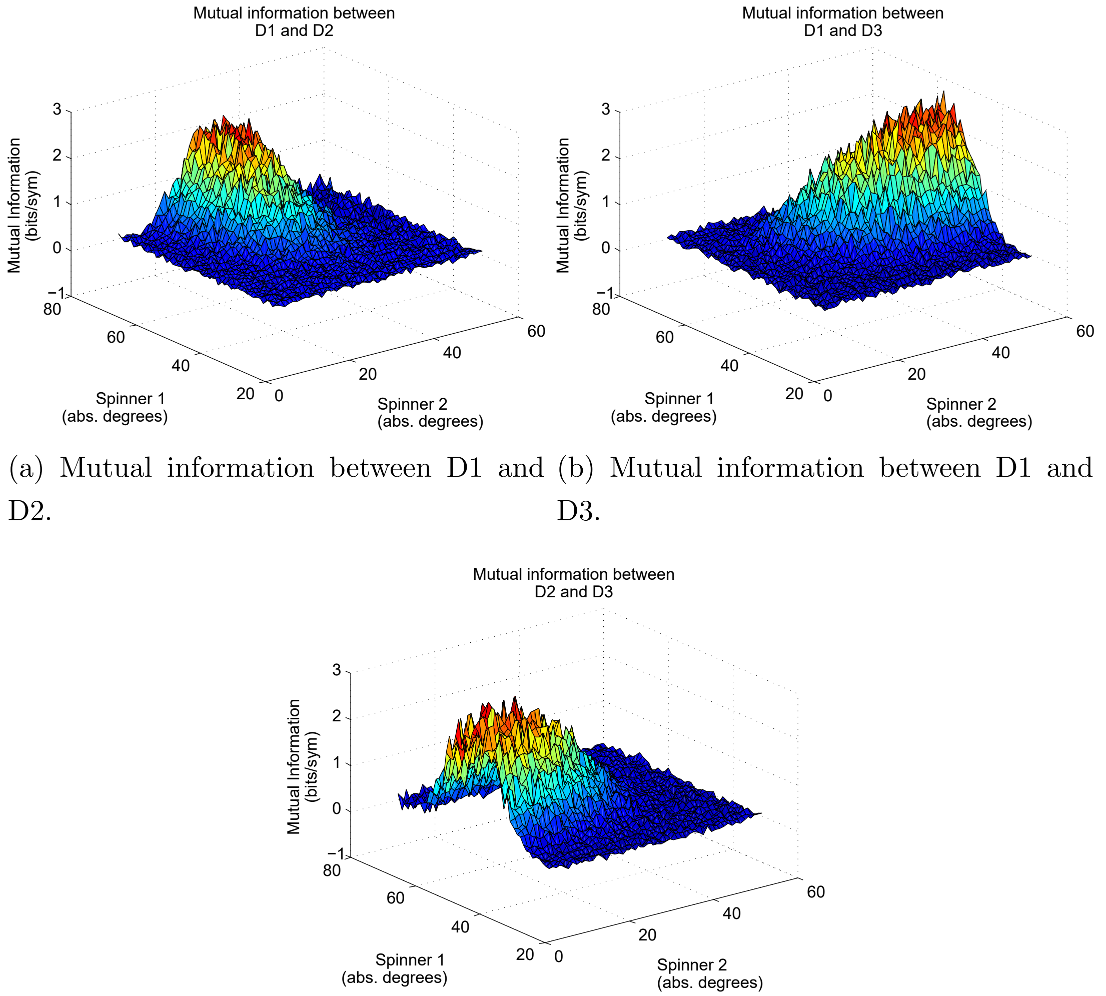
This means that instead of the reading in degrees of spinner S2 on the x-axis, there is instead the intensity received by detector D2. Similarly, the y-axis is now intensity at detector D3 instead of degrees at spinner S1. (Figure 5.13). These



(a) Coherent Light

(b) Thermal light

Figure 5.10: Raw data from two detectors plotted against each other for coherent and thermal light under the same conditions. The histograms show how the density of points is distributed across the voltage range for each detector. The points highlighted in blue are those used to calculate $V_{B|A}$.

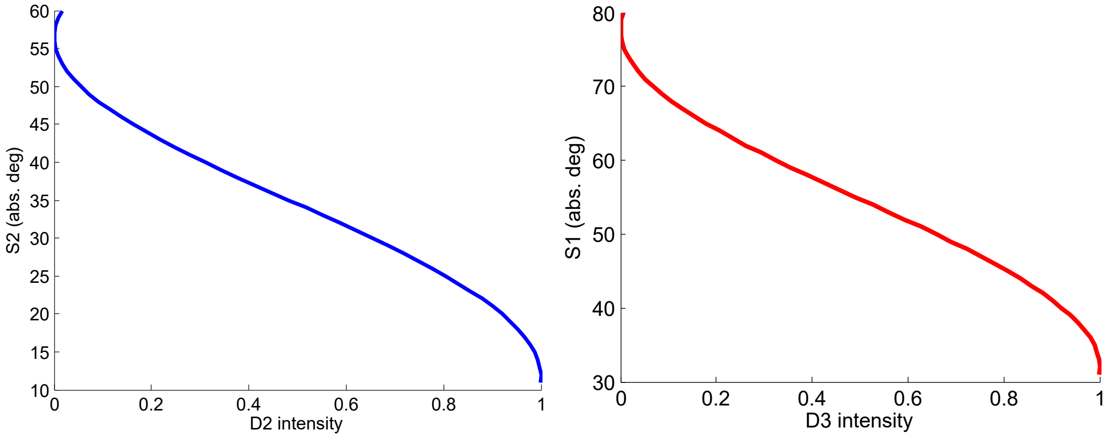


(a) Mutual information between D1 and D2.

(b) Mutual information between D1 and D3.

(c) Mutual information between D2 and D3.

Figure 5.11: Mutual information in the central broadcast set-up using thermal light.



(a) Spinner S2 and detector D2

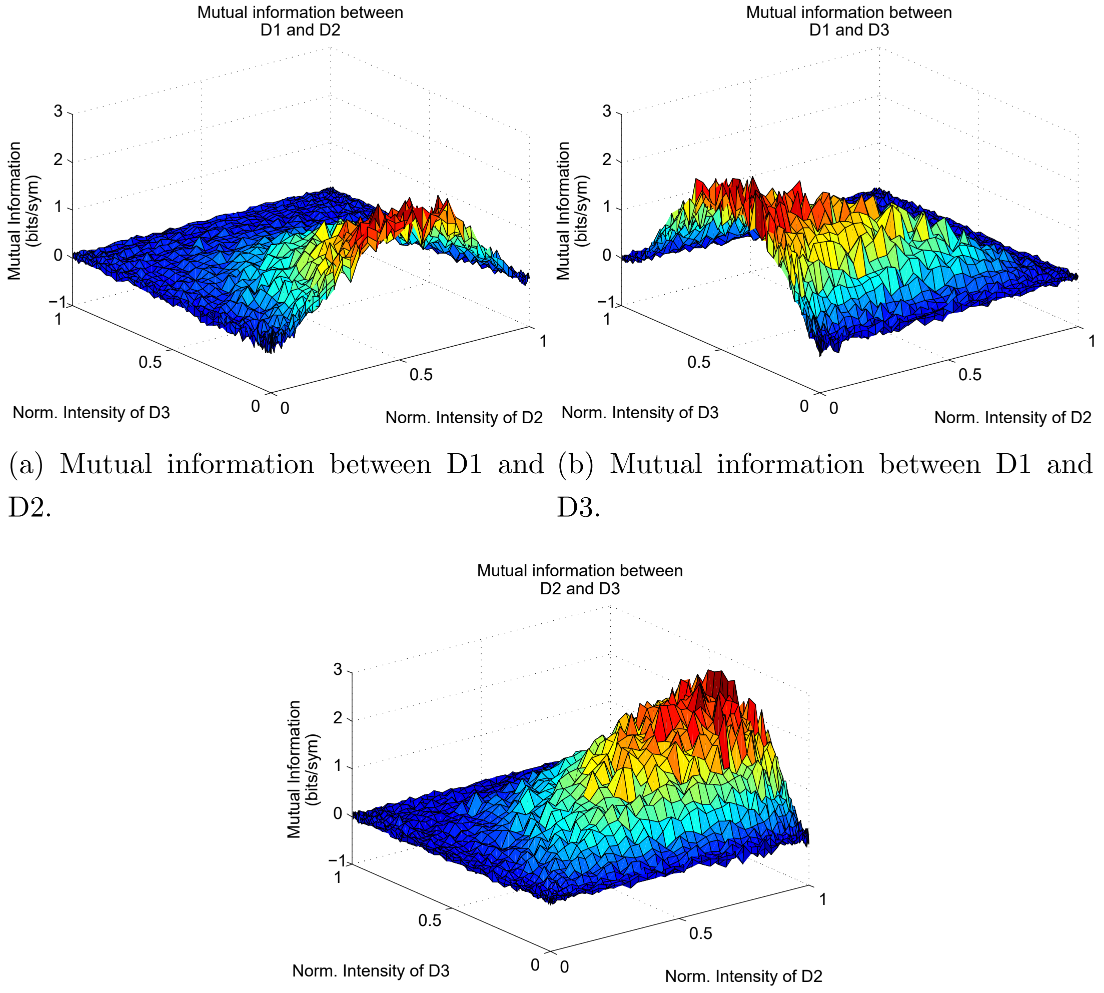
(b) Spinner S1 and detector D3

Figure 5.12: Relationship between spinner position and detector intensity.

intensities have been normalised. It is important to note that even in the case of the graph being a plot of mutual information between D1 and D2 (figure 5.13a), the axis are still the intensity of D2 and D3. This is because, as always, we are interested in the balance of information between all three parties.

As you would expect, the mutual information is highest when the light is split evenly between the two parties involved in the mutual information, with the minimum amount of light arriving at the third party. Now that we know the mutual information between each of the three parties, we can use this to calculate the secret key rate (as discussed in section 5.1.3). First, we look at the most commonly used measure of key rate, $\Delta I = (\mathrm{I(A;B)} - \mathrm{I(A;E)})$. Figure 5.14 shows this key rate in the thermal case for both protocol 1 and protocol 2 (section 5.1.4).

For both of these protocols, there is an area of strong, positive key rate when the beam is split 50:50 between Alice and Bob, and Eve receives very little. In protocol 1, Alice an Bob do worst when Eve has the majority of the beam, but when what is remaining is still split evenly between them. Interestingly, they look better if the remainder is split unevenly between them. However, perhaps somewhat unexpectedly, in protocol 2, Alice and Bob suffer most when Alice has the majority of the beam.

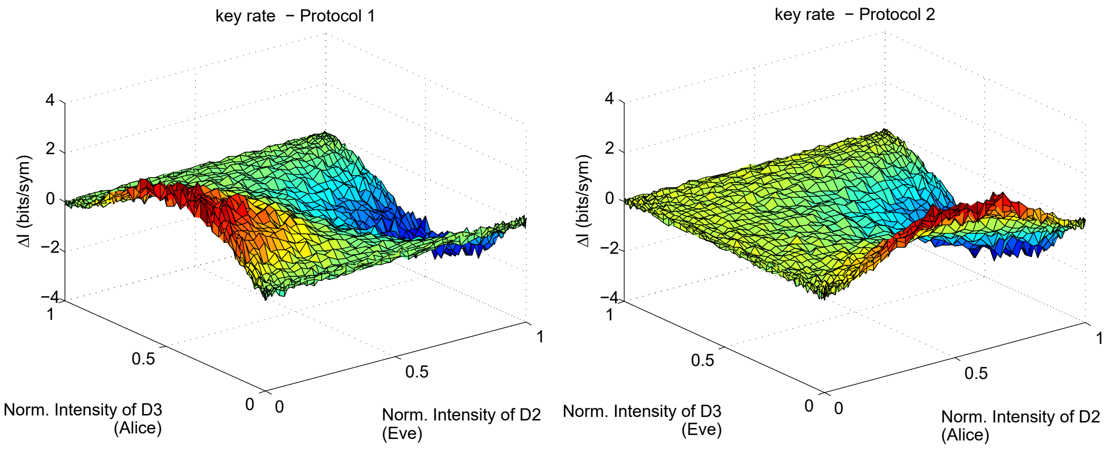


(a) Mutual information between D1 and D2.

(b) Mutual information between D1 and D3.

(c) Mutual information between D2 and D3.

Figure 5.13: Mutual information in the central broadcast set-up using thermal light with the x and y axis changed to normalised intensity at detectors D2 and D3 respectively.



(a) Protocol 1.

(b) Protocol 2.

Figure 5.14: Secret key rate $\Delta I = (\mathrm{I(A;B)} - \mathrm{I(A;E)})$, for both protocol 1 and protocol 2.

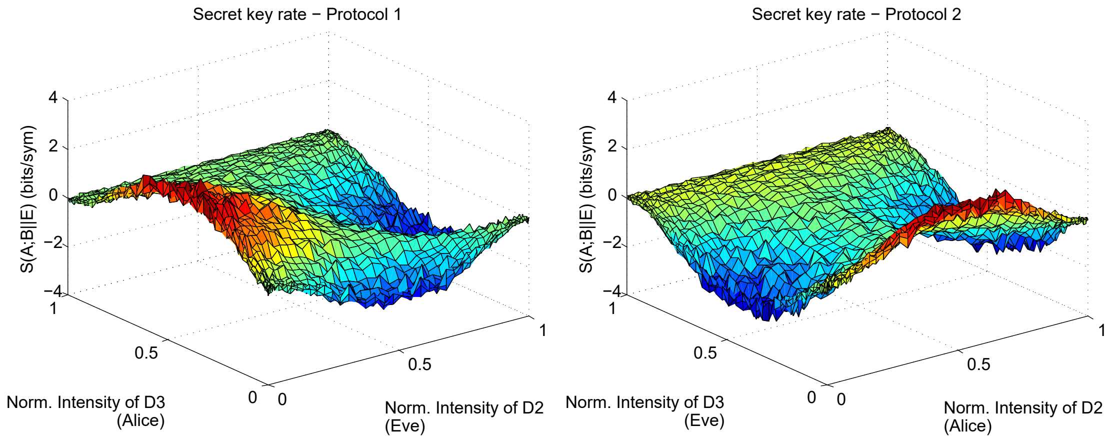
We also look at the secret key rate, $\mathrm{S(A;B||E)} \geq \max[\,\mathrm{I(A;B)} - \mathrm{I(A;E)},\ \mathrm{I(A;B)} - \mathrm{I(B;E)}\,]$, as defined by Maurer (equation 5.1). This is shown in figure 5.15.

Here, the optimal areas remain the same as the previous definition of key rate (ΔI), but additional valleys have appeared. It is easy to see where these come from when taking into account the mutual information figures 5.13, but a little more difficult to interpret.

As before, in protocol 1, there is a valley when Eve receives the majority of the information and the rest is split 50:50 between Alice and Bob. The additional dip occurs when Alice receives almost no information, and the remainder is split equally between Bob and Eve.

In protocol 2, the two valleys have a pleasing symmetry, occurring when Eve has 50% of the beam, and the remaining 50% goes entirely to either Alice or Bob.

It is interesting that while both protocols have the same positive key rate in the same place, the areas of negative key rate, while of the same volume, occur



(a) Protocol 1.

(b) Protocol 2.

Figure 5.15: Secret key rate $\mathrm{S(A;B||E)} \geq \max[\ \mathrm{I(A;B)} - \mathrm{I(A;E)},\ \ \mathrm{I(A;B)} - \mathrm{I(B;E)}\ ]$, for both protocol 1 and protocol 2.

in different places depending on the protocol used.

5.2 Conclusions

While these graphs may look as if they have large areas of negative key rate, it is worth bearing in mind that this is purely due to the statistics of the light beam itself. We have applied no modulation, and no post processing. This would in fact suggest that we may be able to enhance secrecy in some regimes once these have been applied purely by using a different light source. It is possible however, that security may be reduced in other areas. This emphasises the fact that knowing how much noise is on the channel is essential. If Alice and Bob were unknowingly operating in a regime where Eve had the advantage, then that would be a problem, as the eavesdropper could use the natural correlations in the light to obtain some knowledge of the key.

Chapter 6

Classical Post-processing

6.1 Discretisation

6.1.1 Simulations

In order to see how much of an effect the choice of discretisation method has, a simulated channel was constructed, the output of which was then discretised through a number of different methods. We looked at three different methods of bin sizing and positioning, as well as three different ways of numbering them for between two and 128 bins.

6.1.2 Methods

We started by attempting to reproduce the channel described in [82], but were unable to produce a reasonable output using the information given. Instead, we simulated a simple channel which operated as follows.

Alice starts with a Gaussian distribution of values, representing her detector voltage readings, with mean $\mu = 0$, and a standard deviation[1] $V_A = 100$. This is 10,000 elements long. Bob's values are then produced by a combination of Alice's

[1] MATLAB uses standard deviation in it's normrnd function rather than the usual variance.

values, A, the channel transmission, T, and some noise.

$$B = T(A + V_P) + V_{D_B}, \tag{6.1}$$

where V_P is another Gaussian distribution of values, representing preparation noise, and any noise on the channel that occurs before Eve's wiretap[1] (any channel noise common to both Bob and Eve). V_{D_B} is again, a Gaussian, and this one represents any noise only Bob has, for example any detector noise.

In this case, we treated Eve exactly the same as Bob.

$$E = (1 - T)(A + V_P) + V_{D_E}, \tag{6.2}$$

except she receives the inverse proportion of the original signal sent by Alice. This is what happens in a wire-tap model. She also has her own, independent, detector noise, V_{D_E}. While this is a very simplistic channel, it has a lot of scope for modification. In our example, for each of V_P, V_{D_B} and V_{D_E}, the mean was 0 and the standard deviation was 1. These, along with $V_A = 100$ were chosen more or less arbitrarily. It was found that they gave mutual information graphs much like those found in the experimental data of chapter 5, and that pictured in [82]. Whilst experimenting with different noise levels, they acted as expected. The only thing that would increase the similarity to the experimental data would be to maybe add a third source of noise, common to Bob and Eve, but independent of the channel transmission. This would enable the expected reduction in mutual information with increased noise, whilst preventing the mutual information between B and E getting too variable.

The mutual information before discretisation was found in the same way as in chapter 5, using equation 5.2[2].
The data for each party was then normalised so that the mean lay at zero before

[1] Of course, QKD protocols are not limited to wiretap attacks, but this is an easy way to visualise what is happening.

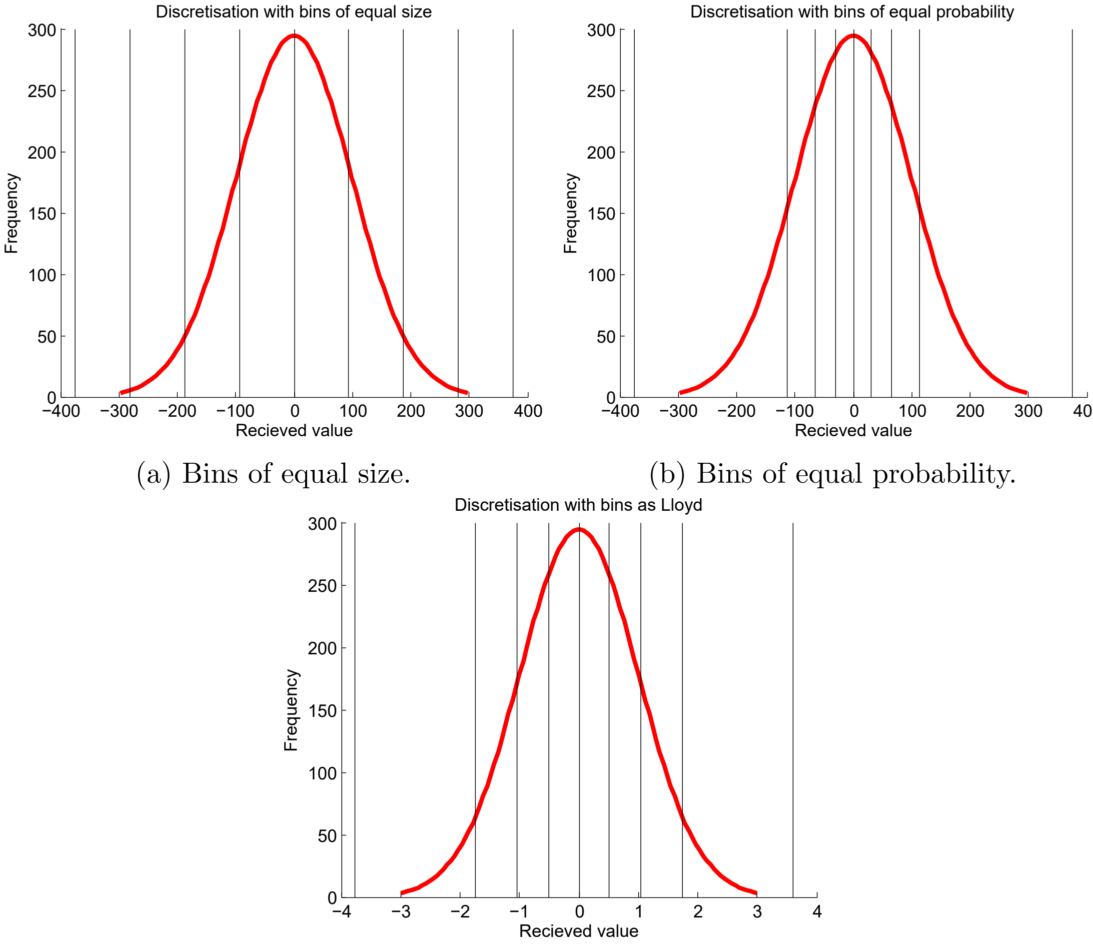
[2] All the code for this can be found on the supplementary CD

the discretisation.

The three bin positioning methods looked at were: placing bins at equal size, equal probability and as described by Lloyd in [93]. Bins of equal size are spaced equally along the x-axis, with fewer points falling into the bins at each extreme. Bins of equal probability have the same number of points in each bin, and are spaced unevenly on the x-axis. Lloyd's bins are designed to minimise the number of least squared errors between Alice and Bob. These are shown in figure 6.1.

Lloyd's algorithm is an iterative process designed to find the bin positions where the centre of mass of data within each bin is as close to central as possible. It's a simple repeated sequence of steps that proceeds as follows. First, a temporary set of bins are added as a start point. In our case, we used bins of equal spacing. Then the centre of mass of the data points in each of the bins is found. These centres of mass are unlikely to be equidistant along the x-axis from the two sides of the bin. These bin edges are moved to ensure this is the case. Of course, now the centre of mass will need recalculating. This continues until the bin edges stop moving on each iteration. The beauty of this method is that if done on a Gaussian set of data which has been normalised so the standard deviation is one, the bin edges will remain in the same position for any Gaussian which has been through that same normalisation. This means you only need to calculate the bin edges once. Indeed, in his paper [93], he gives a table of bin edges up to 16 bins. We built a copy of his algorithm (which can be found on the supplementary CD) which produced the same results up to that limit, and then extended the calculations to 128 bins.

For each of these bin positioning methods, the data was first normalised so that it was centred on zero. This zero point then became the divider between two bins. This was done to help with computing power, (you then only need to calculate the bins for one side of the Gaussian then flip them across). Whether this restriction has any effect on the mutual information or secret key rate is still an open question.



(a) Bins of equal size.

(b) Bins of equal probability.

(c) Bins using Lloyd's algorithm.

Figure 6.1: The bin positioning systems used during discretisation of the data. The x-axis on (c) is different, as the width of data entering Lloyd's algorithm is normalised to have a standard deviation of one.

Binary			Gray			FLFSR		
0	0	0	0	0	0	0	0	1
0	0	1	0	0	1	1	0	0
0	1	0	0	1	1	0	1	0
0	1	1	0	1	0	1	0	1
1	0	0	1	1	0	1	1	0
1	0	1	1	1	1	1	1	1
1	1	0	1	0	1	0	1	1
1	1	1	1	0	0	0	0	1

Figure 6.2: Binary numbering systems.

The three numbering schemes used are normal binary numbering, a Gray code, where successive numbers only differ by one bit, and a Fibonacci linear feedback shift register (F-LFSR), which produces pseudo-random numbers. Examples of these are shown in table 6.2.

Normal binary numbers, are similar to the decimal notation we use every day, in that bits of increasing significance increase incrementally. So, first the least significant bit increases, then once that's completed a full round, the next significant bit increases, and the least significant completes another round, and so on.

Gray codes are really awesome as they only differ by one bit between each successive value. This means that they can minimise the effect of noise, as transmitted points with a small amount of noise on will only result in one bit difference in the resulting bit strings. There are a few ways of constructing these, but the one used here relied on flipping successive sections of bits. If you start with the first two values the same as you would in binary, then instead of repeating the least significant bits with an increase in the next significant bit, the pair of least significant bits are flipped. This then repeats with the next four bits, then eight, and so on. This is shown in table 6.3.

0	0	0
0	0	1
0	1	1
0	1	0
1	1	0
1	1	1
1	0	1
1	0	0

Figure 6.3: Formation of a Gray code by flipping blocks of bits.

A Fibonacci linear feedback shift register (FLFSR) takes bits in certain positions, passes them through XOR or XNOR gates, and the resulting bit becomes the first bit in the series, all the others shuffling along a place. Here we used a very simple example of just XORing the final two bits together. This is shown in figure 6.4 It is notable that a bit string made up entirely of zeroes will only produce other bit strings made only of zeroes and thus loses you all your secrecy. So that is best avoided.

During our analysis, all three parties were limited to using the same discretisation method. This is mainly because we are simply investigating whether the method used has an effect on either the mutual information or the secrecy of the key. If such an effect does exist, then is it severe enough to be worth worrying about? What we have built is a simple proof of principle model. The question as to whether this is the most optimal strategy for any of the parties is an interesting one.

It is generally accepted that Alice and Bob will use the same discretisation method. Intuition tells us that for them to maximise the mutual information from their near identical Gaussians, they logically should discretise them in the same way. If Bob were to know what all Alice's values were, then he could probably come up with a better one, but he does not. Even knowing the error rate of the channel can't help them, as it tells them very little about the individual

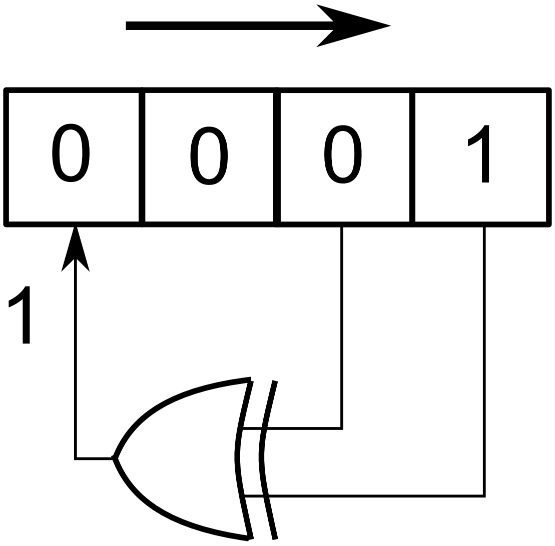


Figure 6.4: A Fibonacci linear feedback shift register.

perturbations on each point.

So if it is generally accepted that the best strategy for Alice and Bob is to use the same method why is it often suggested that Eve should use a different one? One that can, in some mystical way, increase her mutual information with either Alice or Bob? After all, every party in this has almost exactly the same goals - to increase their mutual information. The only difference is that Alice and Bob are also trying to decrease Eve's mutual information, while Eve is unable to interfere with a similar goal with respect to Bob's. While Eve also has the aim or remaining undetected, as the quantum transmission has now finished, and measurements been taken, anything she does now will have no further effect on Alice and Bob's knowledge of her presence.

There are three important things to note here.

1. Alice and Bob have control of this situation. As there is no communication between any of the parties during the discretisation stage, except the prior agreement of which discretisation method to use. Eve is relegated to playing with whatever data she managed to obtain during the original quantum transmission.

2. While it is perfectly possible for there to exist a discretisation method for Eve to use that will dramatically enhance her knowledge of Bob's discretised results, being able to create such a method would require knowledge of those results. Which she cannot have. We have taken for granted here, the fact that Eve does not have complete knowledge of either the transmission Alice sent through the quantum channel, or the quantum transmission received by Bob. Because in either of these cases, the CVQKD scheme will have failed. Without any of this knowledge, she cannot construct a discretisation method that increases her information about Bob's (or Alice's) discretised results except by accident.

   This is analogous to Eve cycling through all possible combinations of ones and zeroes until she hits on one that is the same as the final key.

3. Can Eve create a discretisation method that manages to distil more information from the channel transmission than Alice and Bob? - This is similar to the previous point, but not exactly the same. This argument proceeds as follows: If Bob's discretisation method is not the best one ever, then Eve could make a better one and know more information about the channel transmission.

   To be perfectly honest, yes she probably could. But if Eve has that much information about the channel transmission, then the quantum transmission part of your protocol probably needs rethinking. She should be limited in the amount of information she can possibly gain about this.

6.1.3 Results

Before discretising the data, the mutual information between each pair of parties was calculated. This was done via the same method as in chapter 5, using the variances. A selection of mutual information graphs at different spinner rotations from the central broadcast experiment are shown in figure 6.5. If you label D3 as Alice, D2 as Eve and D1 as Bob, (protocol 2 from that chapter), then the figures are comparable to these simulations. As the value of spinner 1 is increased, less of the beam passes to Alice (D3) and more to Bob and Eve. This increases their

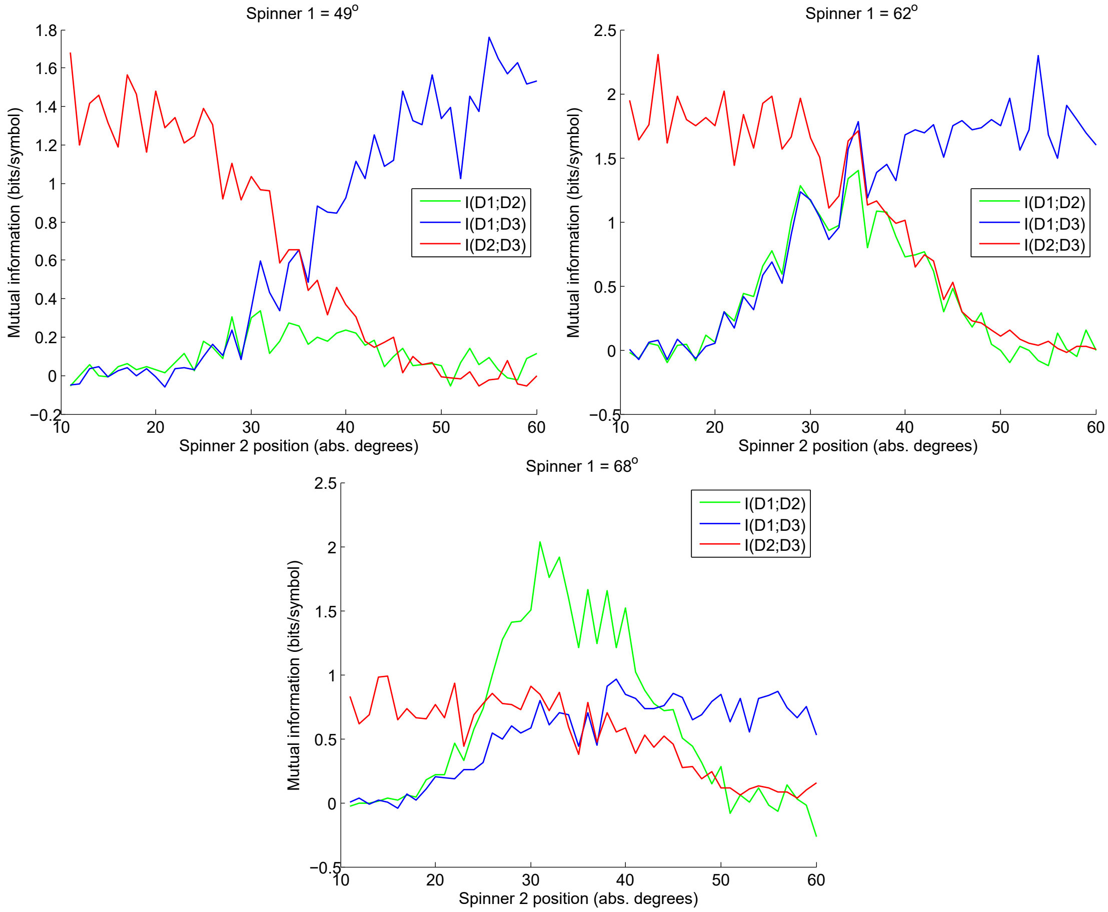


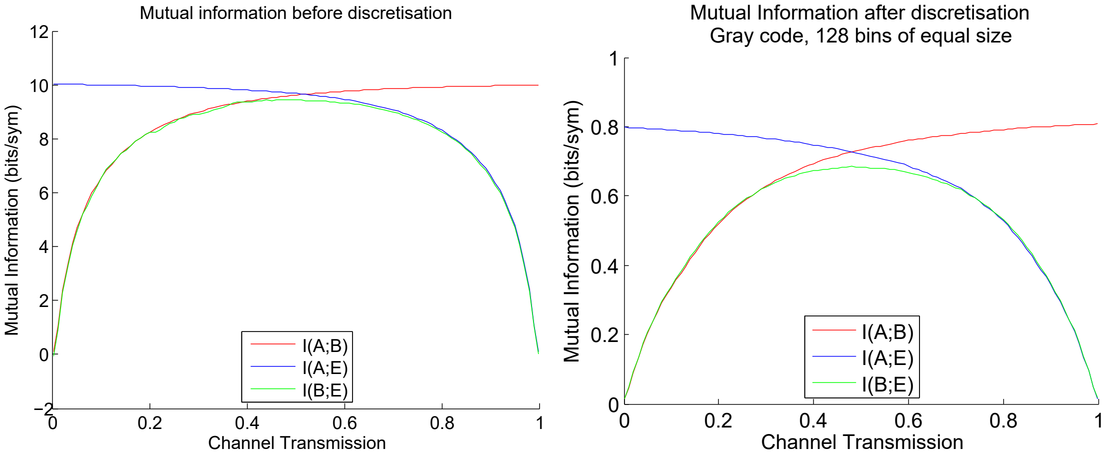
Figure 6.5: The mutual information for each pair of parties from the experimental data from the central broadcast scheme in chapter 5.

mutual information.

The mutual information of the simulation data before discretisation, and an example of what it can look like after discretisation is shown in figure 6.6.

The mutual information after discretisation is shown in figure 6.7 for the nine different discretisation methods.

Bin positioning appears to have a sizeable effect on the mutual information, especially when the data is split into higher numbers of bins. Lloyd's algorithm



(a) The mutual information before discretisation.

(b) The mutual information after discretisation using 128 equally spaced bins and numbered using a Gray code.

Figure 6.6: The mutual information before and after discretisation.

tends to extend higher mutual informations into lower channel transmissions. This makes sense, as at lower channel transmissions, there is more noise. If all the divisions between bins are bunched towards the centre of the Gaussian, as they are with bins of equal probability, then they are surrounded by a higher number of points, so more noise in transmission will cause more errors between Alice and Bob after discretisation. However, with Lloyd's algorithm, where the bins are spaced slightly further apart, there will be fewer noisy bits crossing the bin boundaries, so fewer errors created.

Interestingly, dividing the Gaussian into bins of equal size seems to have a somewhat unpredictable effect. In some cases, it pips Lloyd's algorithm to having the highest mutual information, but in others, it is dramatically the worst. These bad areas are when the data is divided into specific numbers of bins, and are heavily dependant on the numbering method used. For example, when numbering with a Gray code, dividing into 8 bins is noticeably worse than either 4 or 16. Similarly, when using a FLFSR, 16 bins is worse than 8 or 32.

The effect of the different numbering schemes is not inconsiderable, and sim-

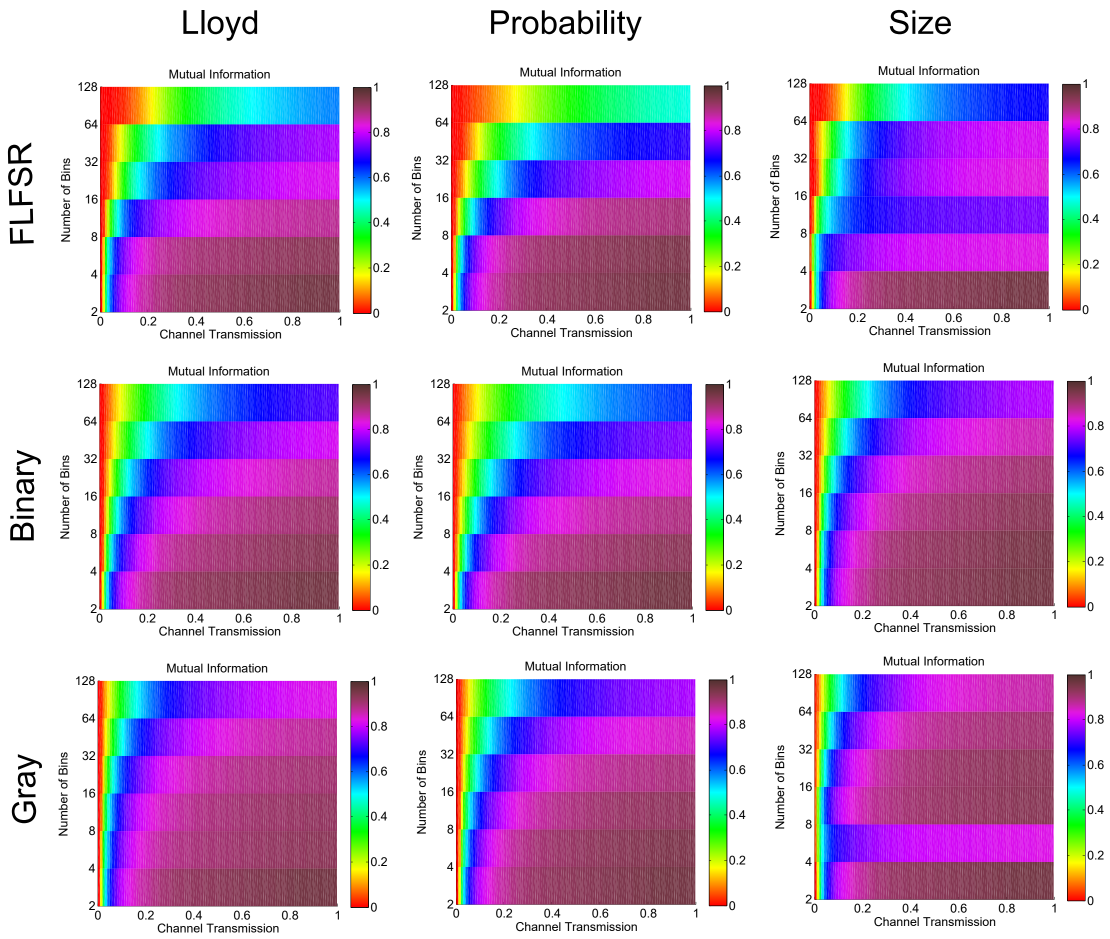


Figure 6.7: The mutual information between Alice and Bob after different discretisation methods. The different columns represent the three different bin positioning methods, and the three rows are the numbering schemes. For each slicing method, two to 128 bins were examined. These are shown by the individual rows of data within each plot. The x axis of each is the channel transmission.

ple to interpret. The pseudo-random FLFSR is the worst of these, as expected, while the Gray code performs the best. The effect of this is to increase the mutual information as you discretise into higher numbers of bins. This is because as you have more bin boundaries, you have more bits with low noise drifting over to adjacent bins. The Gray code minimises the effects of this noise.

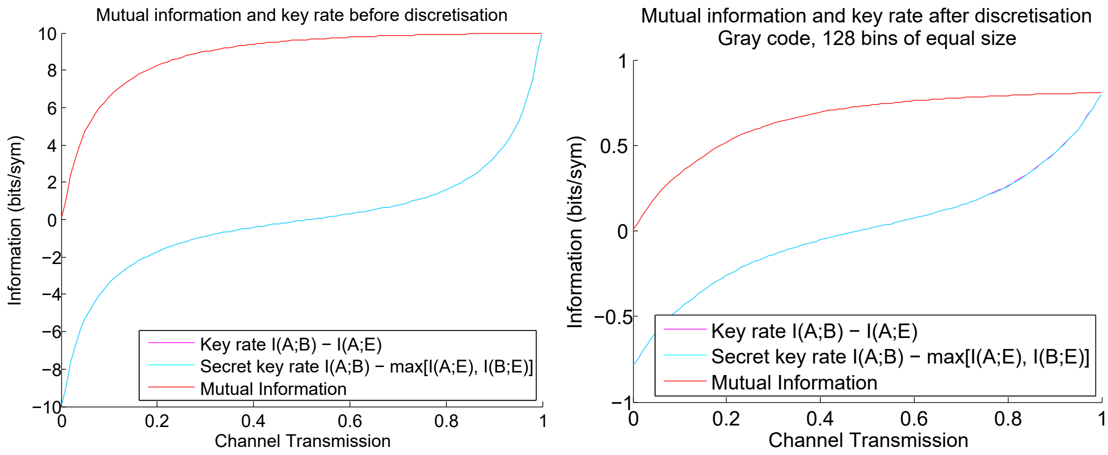
In QKD, a high mutual information between Alice and Bob is not the final goal. Instead, we try to maximise the mutual information between these two parties, whilst minimising the amount of information known by Eve. This can be quantified either by the key rate ($\Delta I = (\mathrm{I(A;B)} - \mathrm{I(A;E)})$), or Maurer's secret key rate ($\mathrm{S(A;B||E)} \geq \max[\ \mathrm{I(A;B)} - \mathrm{I(A;E)},\ \ \mathrm{I(A;B)} - \mathrm{I(B;E)}\ ]$), see section 5.1.3. Generally, you would expect a high mutual information between Alice and Bob, $\mathrm{I(A;B)}$, to lead to a high (secret) key rate, as it appears in both of these quantities.

Values of the mutual information $\mathrm{I(A;B)}$, as well as the key and secret key rates are shown in figure 6.8.

You can see that before discretisation, and for most values of the channel transmission after discretisation, the key and secret key rates remain the same. In fact, you have to look at figure 6.8b quite closely to see the key rate at all. This is because for most of the values of channel transmission, the mutual information between Bob and Eve, $\mathrm{I(B;E)}$, is below that of Alice and Eve, $\mathrm{I(B;E)}$ (Figure 6.6b), so the definitions of the two different key rates become the same.

Due to the small differences between the two rates, and the complexity of the graphs, it is virtually impossible for the human eye to distinguish any difference between the two. For this reason, we only present the secret key rates after discretisation for the nine tested methods, in figure 6.9.

At first glance, the differences between the nine different discretisation methods don't look quite so pronounced for these graphs of secret key rate. This is to be expected, due to the shape of the secret key rate in figure 6.8. The long



(a) The mutual information and key rate before discretisation.

(b) The mutual information and key rate after discretisation using 128 equally spaced bins and numbered using a Gray code.

Figure 6.8: The mutual information and key rate before and after discretisation.

straight middle section pushes all the interesting bits towards the edges.

These graphs all show a similar trend of having a high key rate at high channel transmissions, and a low key rate at lower transmissions. Interestingly the range of channel transmissions for which the key rate is very high (or indeed very low) increases as the number of bins increases. In addition, while the Gray code performs best at extremely high channel transmissions and large numbers of bins, the FLFSR performs much better at middlingly high transmissions. It seems likely that both of these are part of the phenomenon of it sometimes being beneficial for Alice and Bob to purposefully degrade their data in order to distance themselves from Eve.

6.1.4 Relation to the classical

How do these graphs of mutual information compare to the classical measures of mutual information we encountered earlier? Figure 6.10 shows the number of

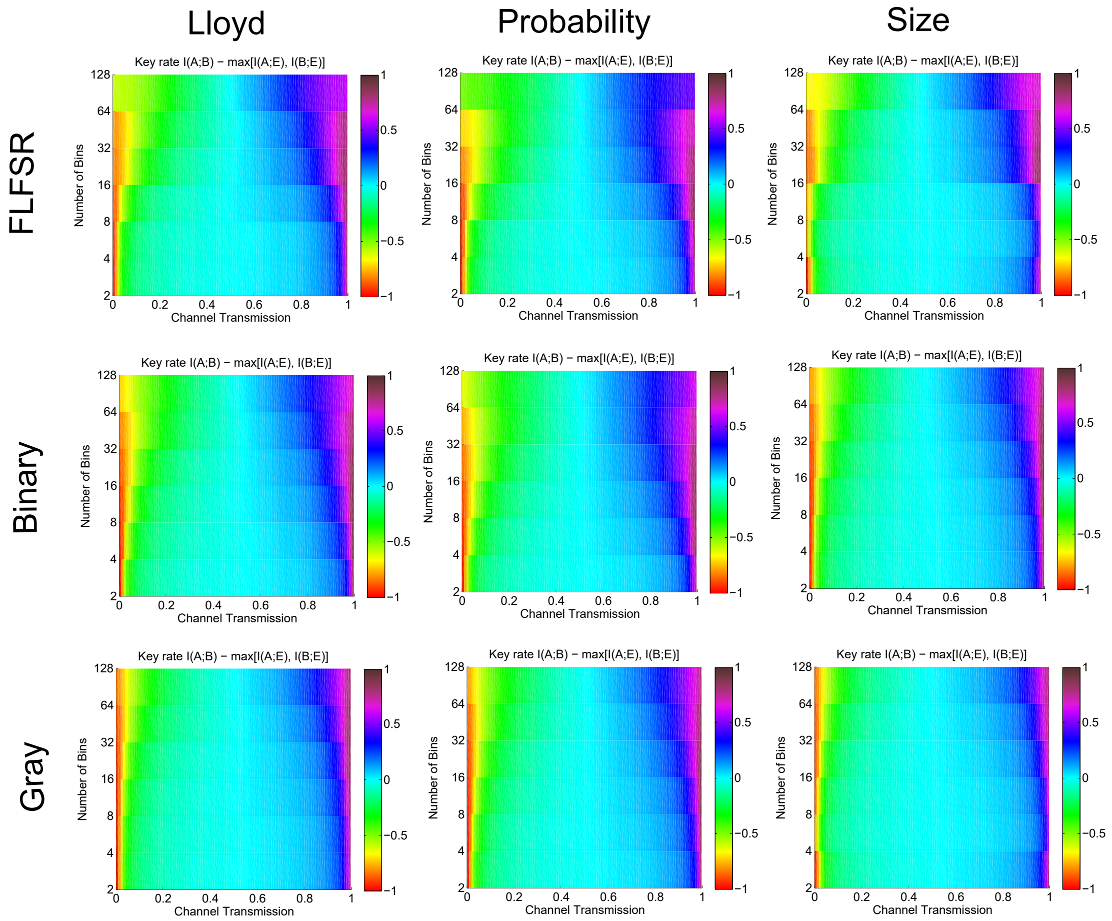


Figure 6.9: The secret key rate $(S(A;B||E) \geq \max[\ I(A;B)-I(A;E),\ \ I(A;B)-I(B;E)\ \ ])$ between Alice and Bob after different discretisation methods. The different columns represent the three different bin positioning methods, and the three rows are the numbering schemes. For each slicing method, two to 128 bins were examined. These are shown by the individual rows of data within each plot. The x axis of each is the channel transmission.

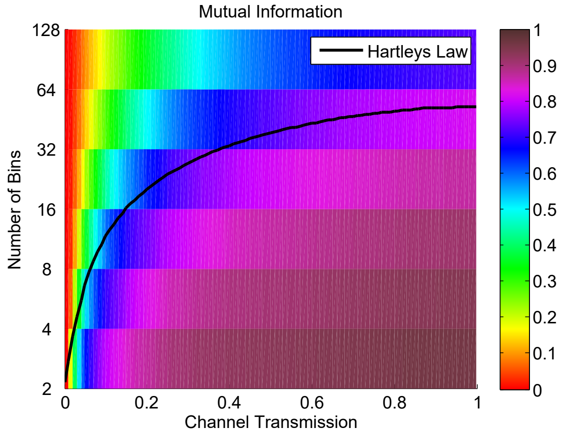


Figure 6.10: The mutual information after discretisation with bins of equal size and a Gray code labelling. On top of this is plotted Hartley's law in black.

bins suggested by Hartley's law.

Hartley's law gives the maximum number of bins it is sensible to use before you start losing information. It does approximately follow the shape of the blue or purple curve in the graph. For maximising the mutual information, this would be a very useful law to keep in mind if you had some idea of your channel transmission.

It is however clear from figure 6.9 that the cases which provide the most mutual information between Alice and Bob are not necessarily the ones which give the highest key rates.

6.1.5 Conclusions

The choice of discretisation method can have a dramatic impact on the mutual information of the resulting bit string. This effect is lessened somewhat in the key rate, and the cases with the highest mutual information are generally not the ones with the highest key rate. Knowing roughly what the channel transmission is before discretisation will help to choose an appropriate discretisation method.

6.2 Reconciliation

6.2.1 Simulations

While there are a number of different variations of cascade, we take a look at the original protocol, and examine eavesdropping techniques which, in unrealistic test cases, can provide the eavesdropper with the entire key. Due to both time constraints, and the fact that this protocol is now more or less obsolete, these techniques have not been tested in more realistic circumstances. It is this author's belief however, that with both time and a more thorough knowledge of computing than this author currently possesses, it would be possible to, at the very least, significantly improve Eve's performance in a realistic scenario using techniques based on those described here. For now, however, it is probably best used perhaps as a cautionary tale.

There are two main approaches we considered, both of which focused on the ability of the eavesdropper to make "soft decisions" about the information available to her, rather than committing to an option which she knows may have some error associated with it.

In QKD, there are three commonly described levels of ability for an eavesdropper. These are known as individual, collective, and coherent attacks.

During an individual attack, Eve is limited to measuring each pulse sent over the quantum transmission between Alice and Bob as it is sent. She commits to the results of that measurement and is unable to change it. She is blind to any further information exchanged between the two legitimate parties, such as during the classical reconciliation stage. This is the weakest level of eavesdropper, and useless for us, as we only concerned with the classical reconciliation at this point.

During a collective attack, Eve is allowed to wait until after Alice and Bob have performed the classical reconciliation steps before she measures (or commits to) the information she's received during the quantum transmission. In the quantum world, she stores all the information from the quantum transmission in a quantum memory, and once she has listened to the classical reconciliation, she can work out an optimal collective measurement to perform on the information held in her quantum memory.

These collective attacks originate from discrete variable QKD [52]. If Alice sends to Bob a whole load of photons with random polarisations and orientations, as in BB84, then Bob receives them and measures them using a random series of polarising filters. Were an eavesdropper to obtain a set of photons where each of which formed an entangled pair with one of Bob's, then a collective attack makes sense. In this scenario, the optimum eavesdropping strategy is to keep all your photons in a quantum memory, and wait for Bob to tell Alice which polarisation filter he used for each measurement. Once Eve has heard this, all she has to do is to apply the same filters to her photons, and due to the entanglement she has an exact replica of Bob's photons, meaning the communication is no longer secure.

In regards to the classical reconciliation in CVQKD, a collective attack enables Eve to make soft rather than hard decisions during every stage of the reconciliation. If information becomes available to her later on in the protocol, she can use this to either improve her confidence in her understanding of the key, or to correct errors she had previously made. This is easily done using a more probabilistic approach, rather than committing to hard zeroes and ones.

For example, if Alice and Bob were to use the not uncommon combination of advantage distillation and cascade, a probabilistic approach can be very useful for Eve. If Alice and Bob proceeded with advantage distillation using a block size of, say, four bits, Eve could use one of five possible outcomes rather then the three Alice and Bob are limited to. Alice and Bob can have an output of zero or one, or they could chose to discard the whole block. Eve can chose zero or one, *or*, depending on her error rate, 0.25, 0.5 or 0.75 (Table 6.11). This means that any information she may receive from cascade can be used to confirm her choices (i.e. rounding 0.25 to 0, or 0.75 to 1). In addition, these soft decisions made here, may help her in cascade, if she knows she has an error in a block, and

Alice			Bob			
Key block	Random No.	XOR	Received	Key block	XOR	Key element
0010	1	1101	1101	0010	1111	1
0100	0	0100	0100	0100	0000	0
1110	1	0001	0001	1010	1011	-
			Eve			
			Received	Key block	XOR	Key element
			1101	0111	1010	0.5
			0100	0110	0010	0.25
			0001	1110	1111	-

Figure 6.11: Eve can use her error rate to make a soft decision about the value of her key element, preventing her from discarding information which may be useful to her later.

the elements in that block are 1, 1, 1 and 0.75, she knows that 0.75 is the most likely element to be in error.

It is possible to argue that Eve doesn't *know*, that that is where her error is, and here is where you have to consider your definition of security. If your definition of security is "Eve can guess the key no better than guessing at random", then this is going to be a problem.

Similarly, during cascade, the parity of blocks of bits are exchanged. The bits are then shuffled, and the parities exchanged again. Eve can work out the probability of error for each bit in a block. As the blocks are shuffled, this probability of error becomes more certain as she compares more and more bits with each other. Using this, along with the few bit values revealed by Alice, Eve can gain a huge amount of information. She can then use this with a form of belief propagation to eliminate large numbers of her own errors.

6.2.2 Coding cascade

Before introducing the eavesdropper, we need a working model of cascade. A detailed description of the protocol is provided here, and the code itself can be found on the supplementary CD.

Alice and Bob start by exchanging and then deleting a small subset of their data to enable them to work out their error rate and, if possible, a bound on the maximum information known by Eve.

Cascade has several rounds. The precise number is determined by Alice and Bob beforehand, and is known to Eve. Generally four is used, as in the original paper [100] Brassard found that in all the cases he tried, this was sufficient to remove all errors between Alice and Bob. This is what we use.

For the first round, Alice and Bob determine a block length based on their error rate via equation 2.13. This ensures that the number of errors per block decreases exponentially with each round. They then divide their key string into blocks of that length.

The original protocol is not hugely clear on what to do with any remainder left between the end of the last whole block, and the end of the string as a whole. In our case, we just turn these into a shorter block, and treat them in the same way as the others. Discarding them is maybe the safer option, but it is worth considering keeping them as a short block, but not exchanging the parity this round. This would enable them to be used as "wild cards" in next round, as Eve would have no clues as to the parity of those particular bits. If these end bits form a large enough fraction of the key as a whole, it is possible they could significantly increase the secrecy of the key. I don't think that the way these bits are treated is necessarily trivial, especially at low error rates between Alice and Bob.

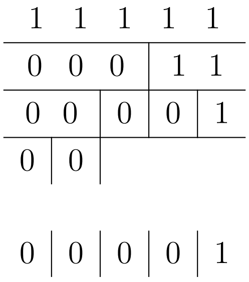


Figure 6.12: The parity and divisions of a block of five bits during cascade. The separate bottom row shows the block of bits, whilst the top section shows the parities as the block is divided into successively shorter sub blocks. The longest half is always towards the left. The parity of the whole block is one, so the first row of the table is populated with ones. The parity of the first (longest) half is 0, so this is populated with 0's. The table continues in this manner.

Once the string is divided into blocks, Alice calculates the parity of each of her blocks and sends this information to Bob. Bob then compares this to the parity of each of his blocks. If the parities don't match, then they enter a process called binary. Here, they each divide each of their blocks in half, and compare the parity of the first half. If this matches, then they know that the error must be in the second half, and they exchange the parity of the third quarter of that block. i.e. the first half of the second half. If this doesn't match, then they take the first half of that half, and so on. I have attempted to describe this visually in figure 2.6.

As the length of the blocks is not limited to 2^n bits, it is not always possible to split the block evenly in half. In this case, we chose the first half of the block to always be larger, and the second half to be the smaller one. This means that a five bit block will end up being divided as shown in figure 6.12.

A point to consider here is that during cascade Alice and Bob reveal the value of any errors they find to the eavesdropper. They don't delete these bits, as they can use the information gained from them to track down the position of further errors. This means that Eve knows the value of those bits. She can also work out the value of the adjacent bit as she knows the parity of the pair. In cases like those in figure 6.12, if Alice and Bob find an error in the first bit of this table, then Eve can also work out the value of a third bit from the information given. She is told the value of the first bit, where Bob has an error. She knows the parity of the pair, so can work out the value of the second bit, but she also knows the parity of the first three bits, so can work out the value of the third bit. Generally, exchanging parities of bits is deemed reasonably secure, as the parities are usually exchanged for pairs of bits. It is worth however, just taking note that when using odd lengths of blocks in this way, you can reveal a bit more than is perhaps obvious at first glance.

Once Alice and Bob have found the locations of their errors, Bob corrects them, changing his bits to match Alice's. Each block now has either no errors in, or an even number of errors. They move on to the nest round of cascade.

In all subsequent rounds of cascade, Alice and Bob start by shuffling the bits in their strings by a random permutation known to all parties. This is also available to Eve. This shuffling is done with the aim of separating out pairs of errors, so that there is more likely to be one error per block again. The block size is twice that in the previous round. This increase in the block size makes it more difficult to find errors, but is more efficient, and reveals less information to Eve.

In our implementation, we shuffle the bits before the beginning of the first round as well, in order to ensure that the errors are evenly distributed throughout the key.

As with the first round, Alice and Bob now use the binary protocol to exchange sets of parities in order to find and correct errors in blocks.

Any errors they find in this round must have a partner error which they shared a block with in the previous round. It is possible that they will find the partner error in this round if it is the only error in its block. (Or, in a block with an odd number of errors). However, if the partner error is in a block with an even number of errors, it may fly under the radar again. To find this error, Alice and

Bob unshuffle the data back to the first round, where the smallest block sizes are used, and use the knowledge of the position of the error that they found, to find the second error. Bob then works out which blocks that error is in in any subsequent rounds - blocks with only one but not both of those located errors will now have an odd error in which can be found. This is repeated until all blocks in all rounds have an even number of errors in. This is the part which gives cascade its name. An example with two rounds of cascade is shown in figure 6.13.

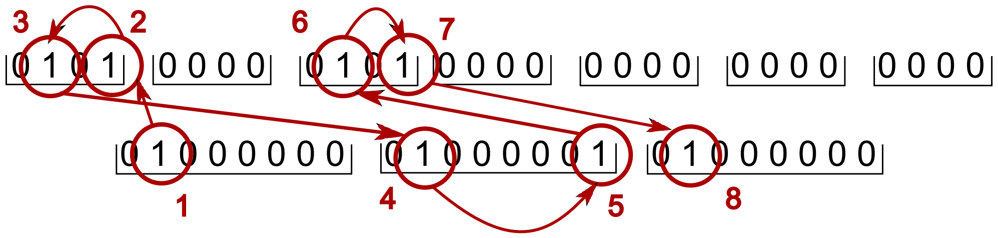


Figure 6.13: Finding errors using the cascading portion of the protocol. Here, errors are shown as a 1, and bits without error 0. As there are an even number of errors in each block in the first round, none are picked up until the second. When Bob finds an error (1), he traces its position back to the first round (2), where he is able to find its partner error (3). This can then be traced forwards again (4) to find a further hidden error (5), which leads to the discovery of more errors (6-8) Without this, only the errors at stages 1 and 8 would have been detected, those at 4 and 5 would have been missed.

6.2.3 Eavesdropping strategies for cascade

All of the information directly revealed in cascade is in the form of parities. The parity of a single bit is its value. Table 6.14 shows an example of a small section of the first round of cascade. For this case, Bob and Eve both had the same error rate of $\frac{1}{6}$. Eve has access to all the information in this table, except Alice and Bob's bit strings in the top two rows.

Alice's String	1	1	1	0	0	0	1	0	1	1	0	1
Bob's String	1	1	1	0	0	1	1	0	0	1	0	1
Eve's String	0	1	1	0	0	0	1	0	1	1	1	1
Alice's Parity	1	1	1	1	1	1	1	1	1	1	1	1
	-	-	-	-	0	0	-	-	0	0	-	-
	-	-	-	-	0	0	-	-	1	-	-	-
Bob's Parity	1	1	1	1	0	0	0	0	0	0	0	0
	-	-	-	-	1	1	-	-	1	1	-	-
	-	-	-	-	0	1	-	-	0	-	-	-
Eve's Parity	0	0	0	0	1	1	1	1	0	0	0	0
	-	-	-	-	0	0	-	-	0	0	-	-
	-	-	-	-	0	0	-	-	1	-	-	-

Figure 6.14: A sample of information directly revealed to Eve during cascade. Bit errors are shown in red. Alice and Bob start by revealing their parities of the first block. Eve compares this to hers and knows she has one error in that block. Alice and Bob have a matching parity, so reveal their parities of the second block. These do not match, so they both reveal their parities of the first half of that block, revealing more information to Eve.

Alice	1	1	1	0	0	0	1	0	1	1	0	1
Bob	1	1	1	0	0	1	1	0	0	1	0	1
Eve	0	1	1	0	0	0	1	0	1	1	1	1
Alice's Parity	1	1	1	1	1	1	1	1	1	1	1	1
	-	-	-	-	0	0	1	1	0	0	1	1
	-	-	-	-	0	0	-	-	1	1	-	-
Bob's Parity	1	1	1	1	0	0	0	0	0	0	0	0
	-	-	-	-	1	1	1	1	1	1	1	1
	-	-	-	-	0	1	-	-	0	1	-	-
Eve's Parity	0	0	0	0	1	1	1	1	0	0	0	0
	-	-	-	-	0	0	1	1	0	0	0	0
	-	-	-	-	0	0	-	-	1	1	-	-
Eve's Parity - Alice's Parity	1	1	1	1	0	0	0	0	1	1	1	1
	-	-	-	-	0	0	0	0	0	0	1	1
	-	-	-	-	0	0	-	-	0	0	-	-

Figure 6.15: Information Eve can directly infer from bits revealed by Alice. Bit errors are shown in Red. Additional information inferred by Eve rather than revealed by Alice, is shown in green.

This is not enough for her to correct any of her errors, but she can confirm that 3 of her bits are correct. From the parity information she is given here, she can conclude a little bit more, shown in figure 6.15. She can also construct an additional row of the difference between her and Alice's parities.

Eve only gains confirmation of one additional bit, bringing her total known bits to four. At this point she knows $\frac{1}{4}$ of the key. If Alice reveals on average two bits of information for every error she and Bob correct in the first round, then the information given to Eve is

$$I_E = b_c \times b_p, \tag{6.3}$$

where b_c is the number of bits corrected, b_p is the number of bits revealed to Eve per bit corrected. If the number of bits corrected in the first round was about half of the total errors, then in our case, where the error rate is $\frac{1}{6}$, Eve would have knowledge of $\frac{1}{6}$ of the key. In practice, more then half of the errors tend to be corrected in the first round (this percentage is dependent on the calculation of initial block length in equation 2.13), and in cases where the block length is an odd number, three bits of information can be revealed when Alice and Bob correct an error.

I will point out here, that this example is chosen for its simplicity to follow. In fact, Brassard suggests [100] that an error rate of less than 0.15 should not be used with cascade because it leaks too much information to the eavesdropper. In our short example, 0.16 was the closest reasonable value to choose to illustrate the technique fully.

Eve knows that when her parity is different from Alice's (when a one occurs in the last three rows of table 6.15), she has an odd number of errors in that block. When their parities match, she has an even number of errors. So in the first block, she has either one or three errors, and no clues as to their location within that block. In the second block, she knows the value of the first two bits. Her second two bits have the same parity as Alice's. This means that she either has no errors in that pair, or they are both errors. Given her error rate (which she knows), she can guess which of these scenarios is more likely. In the third block, she again knows the values of the first two bits. But she knows that she must have an error in the remaining two.

Using all of this information, she can assign a probability of error to each of her bits. All the bits in the first block will have the same probability of error, because she has no information about how any error(s) in that block are distributed. The probability assigned to each of these bits will be the probability that there is one error in that block plus the probability that there are three errors in the block. These are easily calculated from the known error rate. The same kind of thinking can be applied to the two unknown bits in block two, although here the possibility of error is much smaller. In block three, the thinking is much simpler. You know there is one error, and two bits, so the probability of error for each bit is 0.5.

To find the probability of a single bit being in error, we first worked out all of Eve's possible error combinations for the specified block length. The probability of each of these cases occurring, given the error rate, was then worked out using equation 6.4

$$P(\{i\}|e) = \binom{n}{k}(e^k(1-e)^{(n-k)}), \tag{6.4}$$

where $P(\{i\}|e)$ is the probability of a particular bit string $\{i\}$ occurring given the error rate e. The cases of $\{i\}$ which were consistent with the tree of parity differences between Eve and Alice (i.e. the combinations Eve could have for her bits given what she knew about the parities), were then chosen. A normalisation constant was then found, so that all the probabilities of these cases consistent with the parity difference tree added to one. Then for each bit, all the probabilities for the cases with that bit in error were added. This gives us the probabilities found in figure 6.16.

It is reasonably obvious that at this stage, this information doesn't especially help Eve, but after the shuffle at the beginning of the next round, it becomes a lot more useful.

Our current example is not well suited for this next stage, as Bob has already corrected all of his errors, and its a little on the short side.

During this example, Alice and Bob perform two rounds of cascade. Normally they would do four, but this is a short example and neither Bob nor Eve will gain any additional information from the extra rounds in this case. Table 6.17 is the of the same form as the previous tables, but with the parities of each party omitted, only leaving the parity difference between Alice and Eve. The index of each bit is also given.

Here, Eve found one error in the first round, but none in the second. She confirmed the values of six of her bits over the two rounds. Applying what she has learnt from the first round, to the second round, she has found that she has

Alice	1	1	1	0	0	0	1	0	1	1	0	1
Bob	1	1	1	0	0	1	1	0	0	1	0	1
Eve	0	1	1	0	0	0	1	0	1	1	1	1
Alice's Parity	1	1	1	1	1	1	1	1	1	1	1	1
	-	-	-	-	0	0	1	1	0	0	1	1
	-	-	-	-	0	0	-	-	1	1	-	-
Bob's Parity	1	1	1	1	0	0	0	0	0	0	0	0
	-	-	-	-	1	1	1	1	1	1	1	1
	-	-	-	-	0	1	-	-	0	1	-	-
Eve's Parity	0	0	0	0	1	1	1	1	0	0	0	0
	-	-	-	-	0	0	1	1	0	0	0	0
	-	-	-	-	0	0	-	-	1	1	-	-
Eve's Parity - Alice's Parity	1	1	1	1	0	0	0	0	1	1	1	1
	-	-	-	-	0	0	0	0	0	0	1	1
	-	-	-	-	0	0	-	-	0	0	-	-
Eve's probability of error	0.27	0.27	0.27	0.27	0	0	0.19	0.19	0	0	0.5	0.5

Figure 6.16: Eve can use the parities provided to give each of her bits a probability of error.

i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
A	0	0	1	1	0	1	0	1	1	1	1	1	1	0	1	0	1	0	0	1
B	1	0	1	0	0	1	0	1	1	1	1	1	1	0	1	0	0	0	0	1
E	0	0	0	1	1	1	0	1	1	1	1	1	1	0	1	1	1	0	0	1
PE-PA	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1	1	1	-	-
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1	1	-	-	-
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1	0	-	-	-
Er (R1)	0.30	0.30	0.30	0.30	0.30	0.30	0.30	0.30	0.30	0.30	0.30	0.30	0.30	0.30	0.30	1	0	0	0.06	0.06
Err (R2)	0	0.36	0.5	0	0.37	0.37	0.36	0.37	0	0	0.36	0.38	0.38	0.5	0.38	0	0	0	0.36	0.36

i	11	19	7	17	20	5	8	6	10	1	13	2	12	18	15	16	14	3	4	9
A	1	0	0	1	1	0	1	1	1	0	1	0	1	0	1	0	0	1	1	1
B	1	0	0	1	1	0	1	1	1	1	1	0	1	0	1	0	0	1	0	1
E	1	0	0	1	1	1	1	1	1	0	1	0	1	0	1	0	0	0	1	1
PE-PA	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	1	1	1	1	1
	-	-	-	-	-	1	1	1	0	0	-	-	-	-	-	1	1	1	0	0
	-	-	-	-	-	-	-	-	0	0	-	-	-	-	-	-	-	-	0	0
Er (R2)	0.36	0.36	0.36	0	0.36	0.37	0.37	0.37	0	0	0.38	0.38	0.38	0	0.38	0	0.5	0.5	0	0
Err (R1)	0.30	0.06	0.30	0	0.06	0.30	0.30	0.30	0.30	0.30	0.30	0.30	0	0.30	0.30	1	0.30	0.30	0.30	0.30

Figure 6.17: Round one and round two of a cascade example. Red digits are those in error, while green digits are corrected errors.

a 50% chance of error in two of her bits, bits 3 and 14.

At this point, she can apply a technique similar to belief propagation used in LDPC codes (section 2.3.4). First, she takes one of the options, lets start with bit 14. Then she looks at the other round of cascade, and tries to work out what the consequences for that would be if bit 14 was an error. The overall parities for Alice and Eve match for that block in the first round, so if 14 was the error, that block must contain another error. This would have to be in bits 11-13 or 15. Bits 12, 13 and 15 are part of a block in the second round which has a parity of 0. There are two scenarios here. Firstly all of 12, 13 and 15 are errors. This is very unlikely. Secondly, one of 12, 13 and 15 *and* one of 2 or 18 were errors. We know 18 is not an error, which leaves 2. If 2 was an error, this carries on in this vein for some time.

If 11 was also an error, then either 19, 7, or 20 would have to be an error, for the parities to work in round two. We know from round 1 that 19 and 20 are extremely unlikely to be errors. If 7 were an error, either 6 or 8 would also have to be an error, although these at least are plausible. So you would have to make bits 14, 11, 7, and 8 or 6 into errors for the parities to work.

Now we look at what happens if the original error is in bit 3. This would require an additional error in bits 2 or 5. If bit two is an error, we enter back into the mess we had with bits 12, 13 and 15 which we had before. However, if we choose bit 5 to be an error, all the parities would match and nothing further needs to be done. This means we only need to make bits 3 and 5 errors.

Eve chooses to mark bits 3 and 5 as errors, as this brings her closer to the error rate for the key as a whole. It is the most sensible option. In the event of a tie in the number of errors to mark to make the parities match, she can sum the probabilities of error for each relevant bit, and choose the most probable.

In this case, if she corrects those bits, her string is identical to that of Alice and Bob. Unfortunately for Eve, she only *knows* it's identical for $\frac{9}{20}$ cases. It is perfectly possible that there are hidden pairs of errors that she has missed. In some circumstances, this convoluted method will purely confuse her further, although it is probably possible to improve it. Using this technique, Eve will not always know the full key with full confidence. She will however, be able to have a high knowledge with a high confidence.

There are clearly a lot of limitations here. Eve's error rate has to be reasonably low compared to Bob's, but in pen and paper trials[1], this was not as low as you might expect. For example, using the same key length and error rate for Bob as above, and doubling Eve's, on the occasions where cascade found all Bob's errors, Eve had an average of six permutations to try before she was reasonably confident that she had the correct key.

Length is obviously going to be a cause for concern as well. The longer length means more rounds of cascade can be applied (up to four), enabling Eve to harvest more information. It will however likely make belief propagation more difficult.

6.2.4 Conclusions

Cascade is interesting in that shuffling the key every round leaks huge amounts of information to Eve, but is necessary to detect any of Bob's errors which occur in pairs. This is somewhat mitigated by doubling the block size, each time but if Bob finds any errors in subsequent rounds, even more information goes to Eve.

If this project were to be continued, first a working belief propagation algorithm for Eve would need to be constructed. After this, the code would need to be tested with realistic key lengths, and an variety of different eavesdropper error rates. In addition, it may also be interesting to examine how the protocol interacts with advantage distillation, as this will add another layer of error probability to each bit.

[1] We were somewhat hindered by not having completed writing the algorithm.

It would also would be interesting to see if the implementation matters, e.g. if how you treat the remainder when you divide it into blocks, or if you limit the block sizes to being 2^n where n is an integer.

Overall, I would suggest that the general move from cascade to LDPC codes was a very good idea.

Chapter 7

Conclusions and Future work

In chapter 4, we worked towards characterising the light source, which we found could switch smoothly between an ASE regime where it produced thermal light, and a lasing regime where it produced coherent light. This included a lot of detail on attempts to maximise the height of the $g^{(2)}(0)$ peak. This would be most easily improved in future by either purchasing faster detectors or by attempting to reduce the linewidth and increase the coherence length of the source. In addition, it may be possible to reduce the difference in path length between the detectors and the beamsplitter while ensuring a good alignment. This however will be difficult unless the coherence length of the laser is also improved. Another path worth pursuing is the creation of pseudothermal light using the ground glass disks. This is primarily because the light source we have, which switches smoothly between the two regimes is no longer manufactured. Were the pseudothermal light to behave in the same way, and exhibit similar correlations, it may well also demonstrate the increase in secrecy we see in chapter 5. This would be advantageous, as in some situations, for example where lasers are already in use, pseudothermal light may be easier or cheaper to produce. If the correlations produced by pseudothermal light did not exhibit the increased secrecy that we see in thermal light, then that would help to answer fundamental questions about the origin of the extra secrecy, i.e. is it a fundamentally quantum property, or can it be replicated by introducing small fluctuations manually?

In chapter 5, we show that in some circumstances, using thermal light instead of coherent light could increase the secrecy between Alice and Bob. Because this is new, we have started with a small toy model, establishing the basics before building steadily up to what, in time, will hopefully be a full protocol. Where we hope to go from here is reasonably clear. The first steps will be to start applying a modulation and seeing if the use of thermal states still provides an advantage. If it does, will the degree in improvement in secrecy be affected by the length or intensity of the applied modulation relative to the thermal fluctuations? What happens when using pseudothermal light? Does it perform as well (or maybe even better?) than thermal light? This will be a case of testing each new stage thoroughly as the protocol evolves.

Chapter 6.1 looked at the discretisation of Gaussian distributed key elements into a bit string. After simulating a number of different discretisation methods, it was found that the method chosen would have an impact on the key rate. It is perhaps counter-intuitive that this step, which involves no communication, can effectively give information to the eavesdropper if performed poorly, but we have found that this is the case. Here again, we were looking at basic model systems rather than formulating a complete part of a protocol. In order to build on this, it would be necessary to build a more complete channel simulation, better representing the particular system of interest, before working on creating a better discretisation system.

In the final chapter (chapter 6.2), we looked at vulnerabilities in the classical post processing protocols, focusing primarily on cascade. We found that while each stage of these may be considered secure when considered individually, the security reduced considerably when these stages were combined, as they would be in any realistic scenario. This area is both fascinating, and challenging. Unless this receives the attention it requires, it will always be the weakest, most vulnerable part of the system. The most appealing to any real life attacker. This is the part of this thesis (and the field as a whole) where the way forwards is clearest. In this case, completing the attack on cascade will be necessary on order to determine its severity. This is important as, while it has mostly been succeeded by LDPC codes, those too have their disadvantages, and some people are returning to cascade. This is mainly due to the fact that LDPC codes cannot work if I(A;B)<max{I(A;E),I(B;E)}, which can be a problem in some protocols. It should be easily possible to write an improved version of cascade, deleting errors rather than correcting them, limiting block sizes to 2^n where n is an integer, strictly limiting the number of shuffles, and other things which may come to light during the process of a more thorough analysis.

This work starts forming the building blocks which may one day become a fully formed protocol. It is important to study every step in detail, before it can be considered. This includes the classical stages. In addition to this, it is increasingly obvious that while each step may be considered sufficiently secure on its own, that is not necessarily true when combined with additional stages. It is vital that we look at how information leaked during one stage of the protocol can lead to unexpected vulnerabilities in any subsequent stages.

Glossary

In a bid to try prevent confusion, I have compiled a glossary of some of the terms used in this work.

i.i.d

Independent and identically distributed. Used to describe a collection of random variables. Each random variable has the same probability distribution as the others and they are all mutually independent.

Excess noise

Any noise in the channel which is not vacuum noise. Noise which is more than the loss induced noise.

Heterodyne detection

The signal is combined with the local oscillator before measurement. Has also been used to mean Bob splits the beam 50:50 and measures both x and p simultaneously.

Homodyne detection

A special case of heterodyne detection where the signal frequency is the same as the local oscillator frequency. Has also been used to mean Bob randomly chooses whether to measure x or p for each time slot.

Individual attack

Eve is limited to measuring each transmission across the quantum link as it occurs. She cannot use any of the information revealed during the classical post processing procedure.

Collective attack

Eve can keep her states she gains from the quantum transmission until after the whole protocol is complete. She knows all the information transmitted during the classical exchange. She can use this to determine the best basis to measure the states in her quantum memory.

Squeezed light	Noise is reduced in one quadrature at the expense of the other. This preserves the Heisenberg uncertainty principle.
Phase squeezed	Light with a very well defined phase, but large amplitude fluctuations.
Amplitude squeezed	Light with a very well defined amplitude, but large phase fluctuations.
Fock states	Light containing a specific number of photons. Also called photon number sates.
Coherent state	Not strongly related to classical coherence. A pure quantum state corresponding to a single resonator mode. A coherent superposition of Fock states. Usually produced from sources such as a laser.
Vacuum state	A state with a photon number of 0, but still exhibiting quantum fluctuations in the electric and magnetic field.
Spatial coherence	A beam is coherent (has a fixed phase relationship) across the cross section of the beam profile
Temporal coherence	A beam is coherent (has a fixed phase relationship) along its forward travelling length. (ie. same point at different times)
Coherence time	Degree of time over which first order coherence is lost
First order coherence	Light produces an interference effect by interference of the phase in systems such as a Michaelson interferometer or Young's double slit experiment. Better described by maths than with words.
Second order coherence	Light has fluctuations in intensity caused by bunching of photons. Better described mathematically.
Coherence length	The coherence time in units of length
Privacy amplification	Improving the privacy of the key obtained from reconciliation by for example hash functions
Information reconciliation	Another term for error correction. Sometimes used as a general term to describe any classical error correction or advantage distillation etc. protocols occurring after the initial quantum exchange.

Advantage distillation	A technique of maximising the mutual information between Alice and Bob, and reducing that of Eve before entering into reconciliation.
Discrete channel	Inputs and outputs are discrete rather than continuous. For example a finite alphabet.
Memoryless channel	Output does not depend on any previous input.
Intrinsic errors	Errors in the key (before reconciliation) arising due to limited efficiency of quantum protocol, and NOT imperfections of physical set-up or eavesdropping.
Direct reconciliation	During the classical post processing, Bob changes what he has received to match what Alice sent.
Reverse reconciliation	During the classical post processing, Alice changes her values to match what Bob received.
Authenticated public channel	A communications channel to which Eve can listen, but not interfere or pretend to be one of the legitimate participants.

Table 7.1

References

[1] P. Meystre and M. Sargent III. *Elements of quantum optics.* Springer-Verlag, 1991. 3

[2] Oxford Dictionaries. *The Oxford English Dictionary.* Oxford University Press, 2012. 3

[3] J Glauber, Roy. Coherent and Incoherent States of the Radiation Field. *Physical Review*, 131(6):2766, 1963. URL http://0-journals.aps.org.wam.leeds.ac.uk/pr/pdf/10.1103/PhysRev.131.2766. 3

[4] Roy J. Glauber. *Quantum theory of optical coherence : selected papers and lectures.* Weinheim Wiley-VCH, 2007. 3

[5] Rodney Loudon. *The Quantum Theory of Light, 3rd edition.* Oxford University Press, 2000. 4, 16, 17, 19

[6] A. A. Michelson and F. G. Pease. Measurement of the diameter of alpha Orionis with the interferometer. *Proceedings of the National Academy of Sciences*, 7:143–146, 1921. 6

[7] P Goodman, W S Langer, and S P Brumby. The Twiss-Hanbury Brown Controversy : A 40-Years Perspective. 29(2), 1997. 9

[8] J Glauber, Roy. The Quantum Theory of Optical Coherence. *Physical Review*, 130(6), 1963. 9

[9] S. M. Kay and A. Maitland. *Quantum Optics: Proceedings of the tenth session of the Scottish universities summer school in physics 1969.* Academic Press, London and New York, 1970. 9

[10] R Hanbury Brown and R Twiss. Corellation between photons in two coherent beams of light. *Nature*, 177, 1956. 12

[11] Leonard Mandel and Emil Wolf. *Optical Coherence and Quantum Optics*. Cambridge University Press, 1995. ISBN 0521417112. 13, 14, 20

[12] Simon Singh. *The Codebook, The secret history of codes and code-breaking*. Harper Collins, 1999. 27

[13] David Kahn. *The Codebreakers*. Signet, 1973. ISBN 0451089677. 27

[14] Ibrahim A. Al-Kadit. Origins of cryptology: the arab contributions. *Cryptologia*, 16(2):97–126, 1992. ISSN 15581586. doi: 10.1080/0161-119291866801. 27

[15] Ole Immanuel Franksen. *Mr Babbage's secret: The tale of cypher and APL*. Strandbergs Forlag Vedbaek, Denmark, 1984. 27

[16] May 2018. URL `http://wiki.franklinheath.co.uk/index.php/Enigma/Paper_Enigma`. 27

[17] Claude E. Shannon. Communication Theory of Secrecy Systems.pdf. *Bell Labs Technical Journal*, 28(4):657–715, 1949. ISSN 0724-6811. 27, 28, 31, 50

[18] Andrew Steane. Quantum Computing. (July), 1997. ISSN 00344885. doi: 10.1088/0034-4885/61/2/002. URL `http://arxiv.org/abs/quant-ph/9708022{%}0Ahttp://dx.doi.org/10.1088/0034-4885/61/2/002`. 29

[19] T. D. Ladd, F. Jelezko, R. Laflamme, Y. Nakamura, C. Monroe, and J. L. O'Brien. Quantum computers. *Nature*, 464(7285):45–53, 2010. ISSN 00280836. doi: 10.1038/nature08812. URL `http://dx.doi.org/10.1038/nature08812`. 29

[20] Michael A. Nielsen and Isaac L. Chuang. *Quantum computation and quantum information*. Cambridge University Press Cambridge, 2000. 29

[21] Richard P Feynman. Simulating Physics with Computers. 21:467–488, 1982. 29

[22] D. Deutsch. Quantum Theory, the Church-Turing Principle and the Universal Quantum Computer. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 400(1818):97–117, 1985. ISSN 1364-5021. doi: 10.1098/rspa.1985.0070. URL `http://rspa.royalsocietypublishing.org/cgi/doi/10.1098/rspa.1985.0070`. 29

[23] P Benioff. The computer as a physical system: a microscopic quantum mechanical model of computers as represemted by Turing machines. *J.\ Stat.\ Phys.*, 22(5):563–591, 1980. URL `papers3://publication/uuid/BBFEFA7A-FE4E-4026-93F3-8AC1659FA877`. 29

[24] Yuri Manin. *Computable and uncomputable (in Russian)*. Sovetskoye Radio, Moscow, 1980. 29

[25] Ashley Montanaro. Quantum algorithms: an overview. *Nature Publishing Group*, pages 1–8, 2015. ISSN 2056-6387. doi: 10.1038/npjqi.2015.23. URL `http://arxiv.org/abs/1511.04206{%}0Ahttp://dx.doi.org/10.1038/npjqi.2015.23`. 29

[26] Lov K. Grover. A fast quantum mechanical algorithm for database search. *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing - STOC '96*, pages 212–219, 1996. ISSN 07349025. doi: 10.1145/237814.237866. URL `http://portal.acm.org/citation.cfm?doid=237814.237866`. 29

[27] Peter W. Shor. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. pages 124–134, 1995. ISSN 0097-5397. doi: 10.1137/S0097539795293172. URL `http://arxiv.org/abs/quant-ph/9508027{%}0Ahttp://dx.doi.org/10.1137/S0097539795293172`. 29

[28] May 2018. URL `https://www.research.ibm.com/ibm-q/`. 29

[29] May 2018. URL `https://www.microsoft.com/en-gb/quantum/`. 29

[30] May 2018. URL https://research.google.com/pubs/QuantumAI.html. 29

[31] May 2018. URL https://newsroom.intel.com/press-kits/quantum-computing/. 29

[32] May 2018. URL https://quantumcircuits.com/. 29

[33] May 2018. URL https://ionq.co/. 29

[34] May 2018. URL https://rigetti.com/. 29

[35] May 2018. URL https://www.dwavesys.com/home. 29

[36] May 2018. URL https://www-03.ibm.com/press/us/en/pressrelease/53374.wss#release. 29

[37] R. L. Rivest. The md5 message-digest algorithm, April 1992. URL https://tools.ietf.org/html/rfc1321. 30

[38] D. Eastlake and P. Jones. Us secure hash algorithm 1 (sha1), September 2001. URL https://tools.ietf.org/html/rfc3174. 30

[39] Specification for the advanced encryption standard (aes), November 2001. URL https://csrc.nist.gov/csrc/media/publications/fips/197/final/documents/fips-197.pdf. 30

[40] Data encryption standard (des), October 1999. URL https://csrc.nist.gov/csrc/media/publications/fips/46/3/archive/1999-10-25/documents/fips46-3.pdf. 30

[41] Rolf Oppliger. *Contemporary cryptography*. Artech House, 2005. 31

[42] R. L. Rivest, A. Shamir, and L. Adleman. A method of obtaining digital signatures and public key cryptosystems, February 1978. 31

[43] André Chailloux, María Naya-Plasencia, and André Schrottenloher. An Efficient Quantum Collision Search Algorithm and Implications on Symmetric Cryptography. *International Conference on the Theory and Application of Cryptology and Information Security*, Lecture No:211–240, 2017. doi: https://doi.org/10.1007/978-3-319-70697-9_8. 31

[44] May 2018. URL https://csrc.nist.gov/Projects/Post-Quantum-Cryptography. 31

[45] Daniel J. Bernstein and Tanja Lange. Post-quantum cryptography. *Nature*, 549(7671):188–194, 2017. ISSN 0028-0836. doi: 10.1038/nature23461. URL http://www.nature.com/doifinder/10.1038/nature23461. 32

[46] Daniel J. Bernstein, Johannes Buchmann, and Erik Dahmen. *Post-Quantum Cryptography*. Springer-Verlag Berlin Heidelberg, 2009. 32

[47] Stephen Wiesner. Conjugate coding. *ACM SIGACT News*, 15(1):78–88, 1983. ISSN 01635700. doi: 10.1145/1008908.1008920. 33

[48] Charles H Bennett and Gilles Brassard. Quantum Cryptography, Public Key Distribution and Coin Tossing. In *International Conference on Computers, Systems and Signal Processing*, 1984. 33, 34

[49] James L. Park. The concept of transition in quantum mechanics. *Foundations of Physics*, 1(1):23–33, 1970. ISSN 00159018. doi: 10.1007/BF00708652. 37

[50] Vadim Makarov, Andrey Anisimov, and Johannes Skaar. Erratum: Effects of detector efficiency mismatch on security of quantum cryptosystems (Physical Review a (2006) 74 (022313)). *Physical Review A - Atomic, Molecular, and Optical Physics*, 78(1):1–11, 2008. ISSN 10502947. doi: 10.1103/PhysRevA.78.019905. 38

[51] Bing Qi, Chi-Hang Fred Fung, Hoi-Kwong Lo, and Xiongfeng Ma. Time-shift attack in practical quantum cryptosystems. *Quantum Inf. Comput.*, page 9, 2005. ISSN 15337146. URL http://arxiv.org/abs/quant-ph/0512080.

[52] Chi Hang Fred Fung, Bing Qi, Kiyoshi Tamaki, and Hoi Kwong Lo. Phase-remapping attack in practical quantum-key-distribution systems. *Physical Review A - Atomic, Molecular, and Optical Physics*, 75(3):1–12, 2007. ISSN 10502947. doi: 10.1103/PhysRevA.75.032314. 149

[53] Yi Zhao, Chi Hang Fred Fung, Bing Qi, Christine Chen, and Hoi Kwong Lo. Quantum hacking: Experimental demonstration of time-shift attack against practical quantum-key-distribution systems. *Physical Review A - Atomic, Molecular, and Optical Physics*, 78(4):1–5, 2008. ISSN 10502947. doi: 10.1103/PhysRevA.78.042333.

[54] Lars Lydersen, Carlos Wiechers, Christoffer Wittmann, Dominique Elser, Johannes Skaar, and Vadim Makarov. Hacking commercial quantum cryptography systems by tailored bright illumination. *Nature Photon*, 4(686): 5, 2010. ISSN 1749-4885. doi: 10.1038/NPHOTON.2010.214. URL http://arxiv.org/abs/1008.4593. 40

[55] Lars Lydersen, Carlos Wiechers, Christoffer Wittmann, Dominique Elser, Johannes Skaar, and Vadim Makarov. Reply to "Avoiding the Detector Blinding Attack on Quantum Cryptography". *Nature Publishing Group*, 4 (12):1, 2010. ISSN 1749-4885. doi: 10.1038/nphoton.2010.278. URL http://arxiv.org/abs/1012.0476.

[56] Lars Lydersen, Mohsen K. Akhlaghi, a. Hamed Majedi, Johannes Skaar, and Vadim Makarov. Controlling a superconducting nanowire single-photon detector using tailored bright illumination. *New Journal of Physics*, 13, 2011. ISSN 13672630. doi: 10.1088/1367-2630/13/11/113042.

[57] Feihu Xu, Bing Qi, and Hoi Kwong Lo. Experimental demonstration of phase-remapping attack in a practical quantum key distribution system. *New Journal of Physics*, 12, 2010. ISSN 13672630. doi: 10.1088/1367-2630/12/11/113026.

[58] Henning Weier, Harald Krauss, Markus Rau, Martin Fürst, Sebastian Nauerth, and Harald Weinfurter. Quantum eavesdropping without interception: An attack exploiting the dead time of single-photon detectors. *New*

Journal of Physics, 13, 2011. ISSN 13672630. doi: 10.1088/1367-2630/13/7/073024.

[59] Nitin Jain, Christoffer Wittmann, Lars Lydersen, Carlos Wiechers, Dominique Elser, Christoph Marquardt, Vadim Makarov, and Gerd Leuchs. Device Calibration Impacts Security of Quantum Key Distribution. *Physical Review Letters*, 107(11):1–5, 2011. ISSN 00319007. doi: 10.1103/PhysRevLett.107.110501.

[60] Ilja Gerhardt, Qin Liu, Antía Lamas-Linares, Johannes Skaar, Christian Kurtsiefer, and Vadim Makarov. Full-field implementation of a perfect eavesdropper on a quantum cryptography system. *Nature communications*, 2(2027):349, 2011. ISSN 2041-1723. doi: 10.1038/ncomms1348.

[61] Audun Nystad Bugge, Sebastien Sauge, Aina Mardhiyah M Ghazali, Johannes Skaar, Lars Lydersen, and Vadim Makarov. Laser damage helps the eavesdropper in quantum cryptography. *Physical Review Letters*, 112(7):1–5, 2014. ISSN 00319007. doi: 10.1103/PhysRevLett.112.070503. 38

[62] Artur K Ekert. Quantum Cryptography Based on Bell's Theorem. *Physical Review Letters*, 67(6):661–663, 1991. 38

[63] Eleni Diamanti, Hoi-Kwong Lo, Bing Qi, and Zhiliang Yuan. Practical challenges in quantum key distribution. *Nature Publishing Group*, (December 2015):1–12, 2016. ISSN 2056-6387. doi: 10.1038/npjqi.2016.25. URL `http://arxiv.org/abs/1606.05853{%}0Ahttp://dx.doi.org/10.1038/npjqi.2016.25`. 40

[64] Robert Bedington, Juan Miguel Arrazola, and Alexander Ling. Progress in satellite quantum key distribution. *npj Quantum Information*, (March):1–12, 2017. ISSN 2056-6387. doi: 10.1038/s41534-017-0031-5. URL `http://arxiv.org/abs/1707.03613{%}0Ahttp://dx.doi.org/10.1038/s41534-017-0031-5`. 40

[65] May 2018. URL `https://www.idquantique.com/resource-library/quantum-key-distribution/`. 40

[66] May 2018. URL http://qubitekk.com/security/. 40

[67] Sheng Kai Liao, Wen Qi Cai, Johannes Handsteiner, Bo Liu, Juan Yin, Liang Zhang, Dominik Rauch, Matthias Fink, Ji Gang Ren, Wei Yue Liu, Yang Li, Qi Shen, Yuan Cao, Feng Zhi Li, Jian Feng Wang, Yong Mei Huang, Lei Deng, Tao Xi, Lu Ma, Tai Hu, Li Li, Nai Le Liu, Franz Koidl, Peiyuan Wang, Yu Ao Chen, Xiang Bin Wang, Michael Steindorfer, Georg Kirchner, Chao Yang Lu, Rong Shu, Rupert Ursin, Thomas Scheidl, Cheng Zhi Peng, Jian Yu Wang, Anton Zeilinger, and Jian Wei Pan. Satellite-Relayed Intercontinental Quantum Network. *Physical Review Letters*, 120(3):30501, 2018. ISSN 10797114. doi: 10.1103/PhysRevLett.120.030501. URL https://doi.org/10.1103/PhysRevLett.120.030501. 40

[68] Giuseppe Vallone, Daniele Dequal, Marco Tomasin, Francesco Vedovato, Matteo Schiavon, Vincenza Luceri, Giuseppe Bianco, and Paolo Villoresi. Interference at the Single Photon Level Along Satellite-Ground Channels. *Physical Review Letters*, 116(25):1–6, 2016. ISSN 10797114. doi: 10.1103/PhysRevLett.116.253601.

[69] Hideki Takenaka, Alberto Carrasco-Casado, Mikio Fujiwara, Mitsuo Kitamura, Masahide Sasaki, and Morio Toyoshima. Satellite-to-ground quantum-limited communication using a 50-kg-class microsatellite. *Nature Photonics*, 11(8):502–508, 2017. ISSN 17494893. doi: 10.1038/nphoton.2017.107. URL http://dx.doi.org/10.1038/nphoton.2017.107. 40

[70] Charles H Bennett, Unit De Montreal, and N David Mermin. Quantum Cryptography without Bell's Theorem. *Physical Review Letters*, 68(5):557–559, 1992. 40

[71] Timothy C. Ralph. Continuous variable quantum cryptography. *Physical Review A*, 61(1):010303, dec 1999. ISSN 1050-2947. doi: 10.1103/PhysRevA.61.010303. URL http://link.aps.org/doi/10.1103/PhysRevA.61.010303. 40

[72] M. Sasaki, M. Fujiwara, H. Ishizuka, K. W. Klaus, Wakui, M. Takeoka, S. Miki, T. Yamashita, Z. Wang, A. Tanaka, K. Yoshino, Y. Nambu,

S. Takahashi, A. Tajima, A. Tomita, T. Domeki, T. Hasegawa, Y. Sakai, H. Kobayashi, T. Asai, K. Shimizu, T. Tokura, T. Tsurumaru, M. Matsui, T. Honjo, K. Tamaki, H. Takesue, Y. Tokura, J. F. Dynes, A. R. Dixon, A. W. Sharpe, Z. L. Yuan, A. J. Shields, S. Uchikoga, M. Legré, S. Robyr, P. Trinkler, L. Monat, J.-B. Page, G. Ribordy, A. Poppe, A. Allacher, O. Maurhart, T. Länger, M. Peev, and A. Zeilinger. Field test of quantum key distribution in the Tokyo QKD Network. *Optics express*, 19(11):10388, 2011. ISSN 18678211. doi: 10.1007/978-3-642-23635-8_6. 41

[73] D. Stucki, M. Legré, F. Buntschu, B. Clausen, N. Felber, N. Gisin, L. Henzen, P. Junod, G. Litzistorf, P. Monbaron, L. Monat, J. B. Page, D. Perroud, G. Ribordy, A. Rochas, S. Robyr, J. Tavares, R. Thew, P. Trinkler, S. Ventura, R. Voirol, N. Walenta, and H. Zbinden. Long-term performance of the SwissQuantum quantum key distribution network in a field environment. *New Journal of Physics*, 13, 2011. ISSN 13672630. doi: 10.1088/1367-2630/13/12/123001.

[74] Mohsen Razavi. Multiple-Access. 60(10):3071–3079, 2012.

[75] M Peev, C Pacher, R Alléaume, C Barreiro, J Bouda, W Boxleitner, T Debuisschert, E Diamanti, M Dianati, J F Dynes, S Fasel, S Fossier, M Fürst, J-D Gautier, O Gay, N Gisin, P Grangier, A Happe, Y Hasani, M Hentschel, H Hübel, G Humer, T Länger, M Legré, R Lieger, J Lodewyck, T Lorünser, N Lütkenhaus, A Marhold, T Matyus, O Maurhart, L Monat, S Nauerth, J-B Page, A Poppe, E Querasser, G Ribordy, S Robyr, L Salvail, AW Sharpe, A J Shields, D Stucki, M Suda, C Tamas, T Themel, R T Thew, Y Thoma, A Treiber, P Trinkler, R Tualle-Brouri, F Vannel, N Walenta, H Weier, H Weinfurter, I Wimberge, Z L Yuan, H Zbinden, and A Zeilinger. The SECOQC quantum key distribution network in Vienna. *New Journal of Physics*, 075001(11), 2009. 41

[76] Bernd Fröhlich, James F Dynes, Marco Lucamarini, Andrew W Sharpe, Zhiliang Yuan, and Andrew J Shields. A quantum access network. *Nature*, 501(7465):69–72, 2013. ISSN 1476-4687. doi: 10.1038/nature12493. URL `http://www.ncbi.nlm.nih.gov/pubmed/24005413`. 41

[77] Liu Jun Wang, Luo Kan Chen, Lei Ju, Mu Lan Xu, Yong Zhao, Kai Chen, Zeng Bing Chen, Teng Yun Chen, and Jian Wei Pan. Experimental multiplexing of quantum key distribution with classical optical communication. *Applied Physics Letters*, 106(8), 2015. ISSN 00036951. doi: 10.1063/1.4913483.

[78] Bing Qi, Wen Zhu, Li Qian, and Hoi Kwong Lo. Feasibility of quantum key distribution through a dense wavelength division multiplexing network. *New Journal of Physics*, 12, 2010. ISSN 13672630. doi: 10.1088/1367-2630/12/10/103042. 41

[79] Frédéric Grosshans and Philippe Grangier. Reverse reconciliation protocols for quantum cryptography with continuous variables. *http://arxiv.org/abs/quant-ph/0204127*, page 5, 2002. URL `http://arxiv.org/abs/quant-ph/0204127`. 42, 122

[80] Ch Silberhorn, T C Ralph, N Lütkenhaus, and G Leuchs. Continuous variable quantum cryptography: beating the 3 dB loss limit. *Physical review letters*, 89(16):167901, 2002. ISSN 0031-9007. doi: 10.1109/EQEC.2003.1314262. 42

[81] U.M. Maurer. Secret key agreement by public discussion from common information. *IEEE Transactions on Information Theory*, 39(3):733–742, may 1993. ISSN 0018-9448. doi: 10.1109/18.256484. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=256484`. 42, 48, 63, 119

[82] Frédéric Grosshans, Gilles Van Assche, Jerome Wenger, Rosa Brouri, Nicolas J Cerf, and Philippe Grangier. Quantum key distribution using Gaussian-modulated coherent states. *Nature*, 421:238, 2003. 42, 45, 133, 134

[83] Christian Weedbrook, Andrew M Lance, Warwick Bowen, Thomas Symul, Timothy C. Ralph, and Ping Koy Lam. Quantum Cryptography Without

Switching. *Physical Review Letters*, 93(17):170504, oct 2004. ISSN 0031-9007. doi: 10.1103/PhysRevLett.93.170504. URL http://link.aps.org/doi/10.1103/PhysRevLett.93.170504. 45

[84] Andrew M Lance, Thomas Symul, Vikram Sharma, Christian Weedbrook, Timothy C. Ralph, and Ping Koy Lam. No-Switching Quantum Key Distribution using Broadband Modulated Coherent Light. *Physical Review Letters*, 95:180503, apr 2005. doi: 10.1103/PhysRevLett.95.180503. URL http://arxiv.org/abs/quant-ph/0504004. 45

[85] Vladyslav C Usenko and Frédéric Grosshans. Unidimensional continuous-variable quantum key distribution. *Phys. Rev. A*, 92(6):62337, 2015. doi: 10.1103/PhysRevA.92.062337. URL http://link.aps.org/doi/10.1103/PhysRevA.92.062337. 46

[86] Tobias Gehring, Vitus Händchen, Jörg Duhme, Fabian Furrer, Torsten Franz, Christoph Pacher, Reinhard F. Werner, and Roman Schnabel. Implementation of continuous-variable quantum key distribution with composable and one-sided-device-independent security against coherent attacks. *Nature Communications*, 6:8795, 2015. ISSN 2041-1723. doi: 10.1038/ncomms9795. URL http://www.nature.com/doifinder/10.1038/ncomms9795. 46

[87] Xuyang Wang, Wenyuan Liu, Pu Wang, and Yongmin Li. Experimental study on all-fiber-based unidimensional continuous-variable quantum key distribution. *Physical Review A*, 95(6):1–9, 2017. ISSN 24699934. doi: 10.1103/PhysRevA.95.062330. 46

[88] a. D. Wyner. The wiretap channel, 1975. ISSN 15387305. 48

[89] Imre Csiszár and János Körner. Broadcast Channels with Confidential Messages. *IEEE Transactions on Information Theory*, 24(3):339–348, 1978. ISSN 15579654. doi: 10.1109/TIT.1978.1055892. 49

[90] Elizabeth Newton, Anne Ghesquière, Freya L. Wilson, Benjamin T. H. Varcoe, and Martin Moseley. Quantum Secrecy in Thermal States. pages 1–20, 2017. URL http://arxiv.org/abs/1711.06592. 53

[91] Gilles Van Assche, Jean Cardinal, and Nicolas J. Cerf. Reconciliation of a Quantum-Distributed Gaussian Key. *IEEE Transactions on Information Theory*, 50(2):394–400, 2004. ISSN 00189448. doi: 10.1109/TIT.2003.822618. 54

[92] N J Cerf, S. Iblisdir, and G. Van Assche. Cloning and Cryptography with Quantum Continuous Variables. *quant-ph/0107077*, page 8, jul 2001. doi: 10.1140/epjd/e20020025. URL `http://arxiv.org/abs/quant-ph/0107077`. 56

[93] Stuart P. Lloyd. Least Squares Quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982. ISSN 15579654. doi: 10.1109/TIT.1982.1056489. 61, 135

[94] J. Cardinal and G. Van Assche. Construction of a shared secret key using continuous variables. *Proceedings - 2003 IEEE Information Theory Workshop, ITW 2003*, pages 135–138, 2003. doi: 10.1109/ITW.2003.1216713. 62

[95] Naftali Tishby, Fernando C. Pereira, and William Bialek. The information bottleneck method. pages 1–16, 1999. ISSN 0305-5728. doi: 10.1108/eb040537. URL `http://arxiv.org/abs/physics/0004057`. 62

[96] Xiaolin Wu and Philip A Chou. Minimum Conditional Entropy Context Quantization. pages 1–11, 2000. ISSN 21578095. doi: 10.1109/ISIT.2000.866333. URL `http://ieeexplore.ieee.org/document/866333/`. 62

[97] R.M. Gray and D.L. Neuhoff. Quantization. *IEEE Transactions on Information Theory*, 44(6):2325–2383, 1998. ISSN 0018-9448. doi: 10.1109/18.720541. 62

[98] Shengli Liu, Henk C.A. Van Tilborg, and Marten Van Dijk. A practical protocol for advantage distillation and information reconciliation. *Designs, Codes, and Cryptography*, 30(1):39–62, 2003. ISSN 09251022. doi: 10.1023/A:1024755209150. 64

[99] Yang Liu, Teng Yun Chen, Liu Jun Wang, Hao Liang, Guo Liang Shentu, Jian Wang, Ke Cui, Hua Lei Yin, Nai Le Liu, Li Li, Xiongfeng Ma, Jason S. Pelc, M. M. Fejer, Cheng Zhi Peng, Qiang Zhang, and Jian Wei Pan. Experimental measurement-device-independent quantum key distribution. *Physical Review Letters*, 111(13):1–5, 2013. ISSN 00319007. doi: 10.1103/PhysRevLett.111.130502. 66

[100] Gilles Brassard and Louis Salvail. Secret-key reconciliation by public discussion. *Eurocrypt1993*, pages 410–423, 1994. ISSN 1124318X. doi: 10.1007/3-540-48285-7_35. URL http://www.springerlink.com/index/AN9KJUNEBM2FD5AH.pdf. 67, 151, 157

[101] Matthieu Bloch, Andrew Thangaraj, and Steven W. McLaughlin. Efficient Reconciliation of Correlated Continuous Random Variables using LDPC Codes. page 9, 2005. URL http://arxiv.org/abs/cs/0509041. 67

[102] Jun Muramatsu and Tadashi Wadayama. Low-Density Parity-Check Matrices for Coding of Correlated Sources. *IEEE TRANSACTIONS ON INFORMATION THEORY*, 51(51):3645, 2005. 67

[103] Charles H Bennett, Franois Bessette, Gilles Brassard, Louis Salvail, and John Smolin. Experimental quantum cryptography. *Journal of Cryptology*, 5(1):3–28, 1992. ISSN 0933-2790. doi: 10.1007/BF00191318. URL http://link.springer.com/10.1007/BF00191318. 68

[104] Chip Elliott, Alexander Colvin, David Pearson, Oleksiy Pikalo, John Schlafer, and Henry Yeh. Current status of the DARPA Quantum Network. pages 1–12, 2005. ISSN 0277786X. doi: 10.1117/12.606489. URL http://arxiv.org/abs/quant-ph/0503058. 69

[105] David Elkouss, Anthony Leverrier, Romain Aleaume, and Joseph J. Boutros. Efficient reconciliation protocol for discrete-variable quantum key distribution. *IEEE International Symposium on Information Theory - Proceedings*, pages 1879–1883, 2009. ISSN 21578102. doi: 10.1109/ISIT.2009.5205475. 70

[106] David Elkouss, Jesus Martinez-Mateo, and Vicente Martin. Information Reconciliation for Quantum Key Distribution. 0(0):1–14, 2010. ISSN 1533-7146. URL http://arxiv.org/abs/1007.1616. 70

[107] D. Klinc, Jeongseok Ha Jeongseok Ha, S.W. McLaughlin, J. Barros, and Byung-Jae Kwak Byung-Jae Kwak. LDPC for Physical Layer Security. *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE*, pages 1–6, 2009. ISSN 1930-529X. doi: 10.1109/GLOCOM.2009.5426065.

[108] Charles H Bennett, Gilles Brassard, and Jean-Marc Robert. Privacy Amplification by Public Discussion. *SIAM Journal on Computing*, 17 (2):210–229, apr 1988. ISSN 0097-5397. doi: 10.1137/0217014. URL http://epubs.siam.org/doi/abs/10.1137/0217014. 73, 74

[109] Anthony Leverrier, Romain Alléaume, Joseph Boutros, Gilles Zémor, and Philippe Grangier. Multidimensional reconciliation for continuous-variable quantum key distribution. *IEEE International Symposium on Information Theory - Proceedings*, pages 1020–1024, 2008. ISSN 21578101. doi: 10.1109/ISIT.2008.4595141. 73

[110] Charles H Bennett, Gilles Brassard, Claude Crkpeau, Ueli M Maurer, and Senior Member. Generalized privacy amplification. *Information Theory, IEEE Transactions on*, 41(6):1915–1923, 1995. ISSN 0018-9448. doi: 10.1109/18.476316. URL http://ieeexplore.ieee.org/xpls/abs{_}all.jsp?arnumber=476316. 74

[111] U Maurer and S Wolf. Privacy amplification secure against active adversaries. *Advances in Cryptology - CRYPTO 1997*, 1294(0):307–321, 1997. doi: 10.1007/BFb0052244. URL http://www.springerlink.com/content/an14788871730828/. 74

[112] Michael Leung. Pyapt, January 2017. URL https://github.com/mcleu/PyAPT. 85

[113] F. T. Arecchi. Measurement of the statistical distribution of Gaussian and laser sources. *Physics Review Letters*, 15(24):912–916,

1965. URL http://0-journals.aps.org.wam.leeds.ac.uk/prl/pdf/10.1103/PhysRevLett.15.912. 110

[114] P Koczyk, P Wiewior, and C Radzewicz. Photon counting statistics - Undergraduate experiment. *American Journal of Physics*, 64: 240–245, 1996. URL http://0-scitation.aip.org.wam.leeds.ac.uk/docserver/fulltext/aapt/journal/ajp/64/3/1.18211.pdf?expires=1413450459{&}id=id{&}accname=2107995{&}checksum=248A30B379DB2B6D3DE1D2D633B1C65E.

[115] M. L. Martinez Ricci, J. Mazzaferri, a. V. Bragas, and O. E. Martinez. Photon counting statistics using a digital oscilloscope. *American Journal of Physics*, 75(8):707, 2007. ISSN 00029505. doi: 10.1119/1.2742400. URL http://link.aip.org/link/AJPIAS/v75/i8/p707/s1{&}Agg=doi. 110

[116] Ueli M. Maurer and Stefan Wolf. Unconditionally secure key agreement and the intrinsic conditional information. *IEEE Transactions on Information Theory*, 45(2):499–514, 1999. ISSN 00189448. doi: 10.1109/18.748999. 119, 122