

December 16, 2019

Abstract

1 Introduction

The field of Quantum cryptography is concerned with the exploitation of the laws of quantum mechanics in order to derive secure methods of communication. In this field, classical cryptographic primitives are interwoven with quantum physics in order to make protocols for tasks such as sending a message securely in the presence of an adversary. Perhaps the first person to realize that there was something to be gained from using quantum mechanics for cryptography was Stephen Wiesner, at the time a university professor at Colombia University in New York. His seminal paper, titled "Conjugate Coding", was originally rejected for publication by the IEEE Information Theory. According to some historical accounts it was too far ahead of its time. After all, it was the first paper that described the encoding of information into quantum states and even provided two examples of its potential applications. It wasn't until a decade later that this paper was published by SIGACT news [15]. In hindsight, IEEE Information Theory likely regretted their choice for not publishing this article; quantum cryptography quickly grew into a large field, both in- and outside of academia. Within academia, this has resulted in a zoo of cryptographic protocols and schemes, many of them with large bodies of research towards their optimization and security proofs. Outside of academia, large international companies such as ABN AMRO and Atos have picked up on some of these protocols, and are working on their practical implementations [1, 2].

The protocol that the Dutch bank is aiming to realize is Quantum Key Distribution (QKD). The aim of this protocol is a secure key exchange between two parties, Alice and Bob. This is achieved when, after the protocol, Alice and Bob are the only parties to share some secret key k . The security of this protocol relies on the fact that measurement of a quantum state disturbs it [9], and that any eavesdropper will therefore always introduce some anomaly in the communicated quantum state. Bennett and Brassard were the first to develop a secure QKD protocol in 1984. A couple of years earlier, the same authors together with Breidbart had already developed another quantum cryptographic protocol, Quantum Key Recycling (QKR). QKR is also a protocol with the aim of achieving secure key exchange between two parties. From a quantum mechanical perspective, it relies on the same principles as QKD. The main differences are in the classical domain. In QKR, Alice and Bob can achieve communication in less steps compared to QKD. In recent work [7], classical communication from Bob to Alice was even minimized to just one bit.

An important property of any quantum cryptographic protocol is its equivalence to other protocols. There are multiple ways in which these equivalences can be exploited, for example to prove security. Shor and Preskill employ this method in a publication [14] where they prove security for one protocol that is equivalent to another, indirectly proving security of the latter.

Quantum cryptographic protocols and their respective equivalences are usually given in the formalism of Hilbert spaces. Although experienced quantum physicists will have no problem understanding protocols in this formalism, it fails to expose many of the features of quantum theory in a clear and intuitive way, especially those features that involve the interaction of multiple systems across time and space. Coecke and Kissinger [3] propose a new - strictly diagrammatic - formalism for quantum theory in an aim to address this problem. As their book was released only recently, this diagrammatic notation has yet to be appreciated by many quantum physicists. Given this fact, and the rise in complexity and popularity of quantum cryptographic methods, we propose to build upon the work of Coecke and Kissinger, and apply it to the field of quantum cryptography. The aim is to work out the recently introduced protocols diagrammatically, together with various equivalences, among which those given in [8]. Since the only assumed knowledge is undergraduate level quantum mechanics, we also provide a crash course to the diagrammatic notation. If successful, this research could not only serve as an opportunity for researchers to view their research from a new, potentially refreshing, perspective, but also provide an intuitive, purely diagrammatic, security proof for QKR.

The structure of this work is such that it can be divided up into two parts. Chapter 2 is a fully fletched introduction to the diagrammatic method for a reader who is familiar with quantum mechanics and Dirac notation. The rest of this report ventures into uncharted terrain. Building upon the introduced notation, we will translate findings and protocols from recent papers into diagrams. In chapter 3 we introduce QKD, QKR, and two more protocols, and build their diagrammatic representations. In chapter 4 we will look at equivalences between protocols. Throughout the report we do not shy away from introducing new notation whenever necessary.

2 Diagrammatic Preliminaries

This section gives a concise introduction to the diagrammatic method for a reader who is familiar with undergraduate level quantum mechanics. Note that [3] provides a more in depth introduction into this diagrammatic method, but does not derive most concepts back to Dirac notation.

Appendix A contains all of the concepts discussed in this section as keywords with hyperrefs to the respective subsections.

2.1 From Dirac to Diagrams

2.1.1 States, Effects, and Hermitian Operations

The **ket** is defined as a triangle with its sharp edge down in diagrammatic notation. It can be interpreted as the preparation of a state, in this case ψ . It is referred to as **state** throughout the thesis.

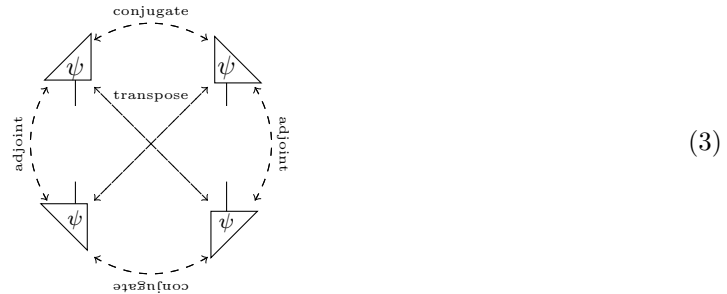
$$|\psi\rangle \equiv \begin{array}{c} | \\ \psi \\ \triangle \end{array} \quad (1)$$

The **bra** in diagrammatic notation is the flipped state, and is referred to as **effect**.

$$\langle\phi| \equiv \begin{array}{c} \triangle \\ \phi \\ | \end{array} \quad (2)$$

Triangles are the smallest building blocks in the diagrammatic notation. Most diagrams can be reduced to just triangles. This makes them a powerful tool for translating complicated diagrams to Dirac notation and vice versa.

From the fact that the **Hermitian adjoint** of a bra gives a ket and reversedly it follows that the operation of flipping a diagram around its horizontal axis corresponds to taking the Hermitian adjoint diagrammatically. Flipping a diagram around its vertical axis is also a legal operation and this corresponds to taking the **Hermitian conjugate**. Both of these operation applied together takes the **transpose**. All of these diagram operations can be summarized as follows:



Equations 1 and 2 show the most general method of writing a bra and a ket in diagrammatic notation. However, the notation also allows for us to take these states and effects further apart. For the state this goes as follows:

$$\begin{array}{|c} \psi \\ \hline \triangle \end{array} \equiv \begin{pmatrix} \psi_0 \\ \psi_1 \\ \vdots \\ \psi_n \end{pmatrix} = \sum_{i=0}^n \psi_i \begin{array}{|c} \text{---} \\ \hline \triangle \\ \text{---} \\ i \end{array} \stackrel{(1)}{=} \sum_{i=0}^n \psi_i |i\rangle \quad (4)$$

The effect is then simply the adjoint of equation 4. Note that the triangle in this equation is not on its side. This is because it is an integer, which is independent of conjugate transformations.

2.1.2 Tensor Product

So far we have seen diagrams with no more than one diagrammatic component but this is soon to change. Therefore, we have to think about what it means for two components to be in one and the same diagram. Having multiple diagrammatic components in the same diagram that are not attached to one another is defined as the **tensor product** over those components. For example, vertically composing a state and effect translates as follows to Dirac notation:

$$\begin{array}{|c} \psi \\ \hline \triangle \\ \psi \\ \hline \triangle \end{array} \equiv |\psi\rangle \otimes \langle\psi| = |\psi\rangle \langle\psi| \quad (5)$$

2.1.3 Wires

The identity map in the diagrammatic notation is given by the following diagram, referred to as a **wire**.

$$| \quad (6)$$

A wire can be reduced to triangles (and subsequently to a ket and a bra) as follows:

$$| \equiv \sum_i \begin{array}{c} \downarrow \\ \triangleup i \\ \triangleup i \\ \uparrow \end{array} \stackrel{(1,2)}{\equiv} \sum_i |i\rangle \langle i| \quad (7)$$

Note that we omit the limits in the summation symbol on purpose. Since we are working with two state systems for the entirety of this report the summation symbol without limits can be interpreted as the sum from 0 to 1 unless explicitly stated otherwise.

To exemplify the wire's behavior, we compose test in- and outputs:

$$\begin{array}{c} \triangleup \phi \\ | \\ \triangleup \phi \\ | \\ \triangleup \psi \end{array} \stackrel{7}{=} \sum_i \begin{array}{c} \triangleup \phi \\ | \\ \triangleup i \\ | \\ \triangleup i \\ | \\ \triangleup \psi \end{array} \equiv \sum_i \langle \phi|i\rangle \langle i|\psi\rangle = \langle \phi|\psi\rangle \quad (8)$$

Furthermore, we can also bend around wires:

$$\begin{array}{c} \text{---} \curvearrowright \text{---} \\ | \end{array} \equiv \sum_i \begin{array}{c} \triangleup i \\ | \end{array} \begin{array}{c} \triangleup i \\ | \end{array} \quad (9)$$

Again composing test in- and outputs:

$$\begin{array}{c} \text{---} \curvearrowright \text{---} \\ | \psi \quad | \phi \end{array} \stackrel{(9)}{=} \sum_i \begin{array}{c} \triangleup i \\ | \psi \end{array} \begin{array}{c} \triangleup i \\ | \phi \end{array} \stackrel{(8)}{=} \sum_i \langle i|\psi\rangle \langle \phi|i\rangle = \langle \phi|\psi\rangle \stackrel{(8)}{=} \begin{array}{c} \triangleup \phi \\ | \psi \end{array} \quad (10)$$

The ϕ thus gets "bent around" along with the wire such that we get back to the situation from equation (8). Intuitively, wires can be seen as information carriers. Every wire has an associated **system type**, the space of the information that it carries. In the context of this report, the system types are generally Hilbert spaces. In section 2.1.8 we will also allow tensor products of Hilbert spaces to be the system types.

2.1.4 Maps

A **linear map** is given by the following diagram:



$$(11)$$

$$\begin{array}{c} \triangle j \quad \triangle k \\ \quad \quad \quad \circ \\ \triangle l \quad \triangle m \end{array} \stackrel{(15)}{=} \sum_i \begin{array}{c} \triangle j \quad \triangle k \\ \downarrow \quad \downarrow \\ \triangle i \quad \triangle i \\ \downarrow \quad \downarrow \\ \triangle l \quad \triangle m \end{array} \stackrel{(8)}{=} \sum_i \langle j|i \rangle \langle k|i \rangle \langle l|i \rangle \langle m|i \rangle = \delta_{jk} \delta_{lm} \delta_{jl} \quad (16)$$

Spiders with single in- or outputs also exist. A spider with a single output creates a random classical variable:

$$\begin{array}{c} \circ \\ \downarrow \\ \triangle i \end{array} \stackrel{(15)}{=} \sum_i \begin{array}{c} \downarrow \\ \triangle i \end{array} = |0\rangle + |1\rangle \quad (17)$$

And a spider with a single input deletes any classical variable $j \in \{0, 1\}$:

$$\begin{array}{c} \circ \\ \downarrow \\ \triangle j \end{array} \stackrel{(15)}{=} \begin{array}{c} \triangle i \\ \downarrow \\ \triangle j \end{array} \stackrel{(8)}{=} \langle 0|j \rangle + \langle 1|j \rangle = 1 \quad (18)$$

An important property of spiders is that they fuse:

$$\begin{array}{c} n \text{ outputs} \\ \dots \\ \circ \\ \dots \\ \circ \\ \dots \\ m \text{ inputs} \end{array} = \begin{array}{c} n \text{ outputs} \\ \dots \\ \circ \\ \dots \\ m \text{ inputs} \end{array} \quad (19)$$

Which can be derived from equation 15

2.1.6 Phase Spiders

Phase spiders follow the same rules as normal spiders. However, perhaps unsurprisingly, they carry a **phase**. This phase is subject to a new set of rules which is evident from the definition of the phase spider.

$$\begin{array}{c} n \text{ outputs} \\ \dots \\ \circ \\ \dots \\ m \text{ inputs} \end{array} \equiv \sum_i e^{i\alpha} \begin{array}{c} n \text{ outputs} \\ \downarrow \quad \downarrow \quad \dots \\ \triangle i \quad \triangle i \quad \dots \\ \downarrow \quad \downarrow \quad \dots \\ m \text{ inputs} \end{array} \quad (20)$$

When two phase spiders meet, they fuse like normal spiders and their phases add:

$$\begin{array}{c} n \text{ outputs} \\ \vdots \\ \text{---} \alpha \text{---} \beta \text{---} \\ \vdots \\ m \text{ inputs} \end{array} \stackrel{(20)}{=} \begin{array}{c} n \text{ outputs} \\ \vdots \\ \text{---} \bigcirc \text{---} \bigcirc \text{---} \\ \vdots \\ m \text{ inputs} \end{array} e^{i\alpha_i} e^{i\beta_i} \stackrel{(19)}{=} e^{i(\alpha_i + \beta_i)} \begin{array}{c} n \text{ outputs} \\ \vdots \\ \text{---} \bigcirc \text{---} \\ \vdots \\ m \text{ inputs} \end{array} \stackrel{(20)}{=} \begin{array}{c} n \text{ outputs} \\ \vdots \\ \text{---} \alpha + \beta \text{---} \\ \vdots \\ m \text{ inputs} \end{array} \quad (21)$$

Furthermore, note that since complex phases get flipped in conjugate transformations, the conjugate of a phase spider is that same spider with an inverted phase:

$$\mathbb{Q}^* \stackrel{(20)}{=} \sum_i (e^{i\alpha_i})^* = \sum_i e^{-i\alpha_i} \stackrel{(20)}{=} \mathbb{Q} \quad (22)$$

2.1.7 Colors and Bases

In the diagrammatic notation, the **color** of an object such as a spider or a triangle determines its basis. For now, we will define two orthonormal **bases**, the Z and X bases. Later on, as more bases will be necessary they will be introduced accordingly. The Z basis has white diagrammatic elements and for the X basis they are gray. The following is an example of how to translate between bases in this notation:

$$\begin{array}{|c|} \hline \downarrow \\ \hline 0 \\ \hline \end{array} = \frac{1}{\sqrt{2}} \left(\begin{array}{|c|} \hline \downarrow \\ \hline 0 \\ \hline \end{array} + \begin{array}{|c|} \hline \downarrow \\ \hline 1 \\ \hline \end{array} \right) \quad (23)$$

$$\begin{array}{|c|} \hline \downarrow \\ \hline 1 \\ \hline \end{array} = \frac{1}{\sqrt{2}} \left(\begin{array}{|c|} \hline \downarrow \\ \hline 0 \\ \hline \end{array} - \begin{array}{|c|} \hline \downarrow \\ \hline 1 \\ \hline \end{array} \right) \quad (24)$$

This equation is of course entirely analogous to its Dirac notation counterpart, with $|0\rangle_x$ and $|1\rangle_x$ being the orthonormal basis states in the X basis and $|0\rangle$ and $|1\rangle$ the orthonormal basis states in the Z basis:

$$|0\rangle_x = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \quad (25)$$

$$|1\rangle_x = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \quad (26)$$

Note that spiders of different colors do not fuse.

2.1.8 Doubling

Doubling is the operation of horizontally composing the conjugate version of a diagram with itself. We already defined previously that having multiple components in the same diagram can be interpreted as the tensor product over those components. Therefore, doubling a diagram is equivalent in Dirac notation to taking the tensor product with its conjugate diagram. In the usual case, where the system type of a single wire is a Hilbert space (\mathcal{H}), the doubled wire has the set of density matrices on that Hilbert space as system type:

$$\mathcal{H} \otimes \mathcal{H} \cong \mathcal{D}(\mathcal{H}) \quad (27)$$

Where $\mathcal{D}(\mathcal{H})$ is the set of density matrices on \mathcal{H} .

Doubled maps and states represent **pure** maps and states and single maps and states represent their **mixed** counterparts. Remember that pure states are those that can be described by a single bra or ket, and can be any superposition of orthonormal basis states. A mixed state, on the other hand, is a statistical ensemble of pure states described by a classical probability distribution over those pure states. The most intuitive interpretation, and how it will mostly be used in this report, is that single diagrammatic components are used in the classical domain whereas doubled components are used in the quantum domain. A doubled state, for example, is a quantum state while a single state represents classical information, potentially on a quantum state. Single wires then carry this classical information and doubled wires carry quantum states.

Doubling in diagrammatic notation is nothing more than placing a second conjugate diagram next to the original diagram. In doubled diagrams lines are drawn thick, a doubled state is thus represented as follows:

$$\begin{array}{c} \mathcal{H} \quad \mathcal{H} \\ \diagdown \quad \diagup \\ \psi \end{array} \equiv \begin{array}{c} \mathcal{D}(\mathcal{H}) \\ \diagdown \quad \diagup \\ \hat{\psi} \end{array} \quad (28)$$

The notation of a state with a hat, such as $\hat{\psi}$, means that that state is pure.

Taking this further apart, we can define an arbitrary density matrix as follows:

$$\sum_{ij} \rho_{ij} \begin{array}{c} \diagup \quad \diagdown \\ i \quad j \end{array} \equiv \begin{array}{c} \diagup \quad \diagdown \\ \rho \end{array} \quad (29)$$

Note that in the case of doubled states, effects, or maps with multiple in- or outputs we need to make sure that the single wires converge correctly:

$$\begin{array}{c} \mathcal{H} \quad \mathcal{H} \quad \mathcal{H} \quad \mathcal{H} \\ \diagdown \quad \diagup \quad \diagdown \quad \diagup \\ f \quad f \end{array} \equiv \begin{array}{c} \mathcal{D}(\mathcal{H}) \quad \mathcal{D}(\mathcal{H}) \\ \diagdown \quad \diagup \\ \hat{f} \\ \diagup \quad \diagdown \\ \mathcal{D}(\mathcal{H}) \quad \mathcal{D}(\mathcal{H}) \end{array} \quad (30)$$

The notation of a map with a hat, such as \hat{f} , means that that map is pure.

Doubled spiders among themselves follow the same rules as normal spiders. That is, they follow the fusion rule from equation 19. However, when a single and double spider meet, they fuse to form one single spider, a so called bastard spider. In diagrams:

$$\begin{array}{c} n \text{ outputs} \\ \vdots \\ \diagdown \quad \diagup \\ \vdots \\ m \text{ inputs} \end{array} = \begin{array}{c} n \text{ outputs} \\ \vdots \\ \diagdown \quad \diagup \\ \vdots \\ m \text{ inputs} \end{array} \quad (31)$$

Which follows from the fact that doubled spiders are two single spiders that each fuse according to equation (19).

2.1.9 Discarding

Discarding is the process of removing part of a diagram from the whole, or removing the whole diagram altogether. For doubled diagrams it is defined as follows:

$$\overline{\text{---}} \equiv \sum_i \triangle_i \stackrel{(28)}{=} \sum_i \triangle_i \triangle_i \stackrel{(9)}{=} \bigcap \stackrel{(9)}{=} \sum_i |i\rangle \langle i| \quad (32)$$

It is trivial to see that applying discarding to any arbitrary (normalized) state always results in the number 1. In fact, discarding a state or map is equivalent to taking its **trace**. As an example, we discard an arbitrary normalized **density matrix**, ρ :

$$\begin{aligned} \overline{\text{---}} \triangle_\rho &\stackrel{(29)}{=} \sum_{ij} \rho_{ij} \triangle_i \triangle_j \stackrel{(10)}{=} \sum_{ij} \rho_{ij} \langle j|i \rangle \\ &= \sum_{ii} \rho_{ii} = \text{Tr}(\rho) = 1 \end{aligned} \quad (33)$$

Discarding is not a pure map. It connects the two counterparts of a doubled state by a single wire. This allows for discarding to be used to purify any arbitrary state or map. For the case of a map Φ , **purification** is as follows:

$$\triangle_\Phi = \overline{\text{---}} \triangle_{\hat{f}} \stackrel{(30)}{=} \triangle_f \triangle_f \quad (34)$$

Note that purification is not only a diagrammatic result, but that it is a commonly used fact in quantum information theory. In general, purification refers to the fact that any mixed state is equivalent to a pure state in a higher dimensional Hilbert space of which we trace a part away. A full proof of this statement can be found in [11], page 110.

Taking the conjugate of the discarding map is actually the preparation of the fully mixed state up to some constant:

$$\overline{\text{---}} \equiv \sum_{i=0} \triangle_i \stackrel{28}{=} \sum_i \triangle_i \triangle_i \stackrel{9}{=} \bigcup \approx \sum_i \frac{1}{2} |i\rangle \langle i| \quad (35)$$

2.1.10 Entanglement

Pure entangled states are those states that are not horizontally separable:

$$\triangle_\psi \neq \triangle_{\psi_1} \triangle_{\psi_2} \quad (36)$$

Since horizontally composed states form the tensor product of those states diagrammatically, this definition is in line with theory, where entangled states are defined as those states that can not be written as the tensor product of two states [11].

An example of such a state is the following:

$$\text{Cup} \stackrel{(9)}{=} \sum_i \text{Triangle}_i \stackrel{(1)}{=} |00\rangle + |11\rangle \quad (37)$$

Which is up to a number the first Bell state, B_0 . Although it may seem like this state separates since it is made up of two triangles, it does not. Both triangles are correlated through the same index. Indeed, if the indices were different for each of the triangles this would not be an entangled state.

2.2 Advanced diagrammatic concepts

2.2.1 The yanking equations

In [3] the authors introduce a couple of equations for bending around wires called the yanking equations. One of them will be used later on in this report, so it will be introduced here:

$$\text{Sine wave} = | \quad (38)$$

We can easily check that this is true by decomposing the wires to triangles as follows:

$$\text{Sine wave} \stackrel{7}{=} \sum_{ij} \text{Triangle}_i \text{Triangle}_j \stackrel{8}{=} \sum_{ij} \delta_{ij} \text{X} = \sum_i \text{X} \stackrel{7}{=} | \quad (39)$$

2.2.2 Basis and phase translations on the Bloch sphere

In section 2.1.7 we already saw how to translate between different bases diagrammatically. The approach was to translate Dirac bra's and kets directly to the appropriate triangles. Using phase spiders, we can employ a more purely diagrammatic method. This stems from the fact that applying a phase to a state corresponds to a rotation on the Bloch sphere. Let's see where a spider without a phase lies on the Bloch sphere:

$$\text{Spider} \stackrel{17}{=} \sum_i \text{Triangle}_i = \text{Triangle}_0 + \text{Triangle}_1 \approx \frac{1}{\sqrt{2}} (\text{Triangle}_0 + \text{Triangle}_1) \stackrel{23}{=} \text{Triangle}_0 \quad (40)$$

On the Bloch sphere in equation 42 we indeed see that the phaseless white spider is on the same location as the gray state 0.

For a white spider with a phase of π we get the following:

$$\textcircled{\pi} \stackrel{17}{=} \sum_i e^{i\alpha_i} \text{triangle}_{\downarrow i} = e^0 \text{triangle}_{\downarrow 0} + e^{i\pi} \text{triangle}_{\downarrow 1} \approx \frac{1}{\sqrt{2}} (\text{triangle}_{\downarrow 0} - \text{triangle}_{\downarrow 1}) \stackrel{24}{=} \text{triangle}_{\downarrow 1} \quad (41)$$

So a white spider with π phase corresponds to the gray state 1.

The gray spiders and white basis have a similar relationship and these are included as well in equation 42.

$$\begin{array}{c} \text{triangle}_{\downarrow 0} \approx \textcircled{0} \\ \text{triangle}_{\downarrow 1} \approx \textcircled{\pi} \end{array} \quad \begin{array}{c} \hat{z} \\ \uparrow \\ \text{circle} \\ \downarrow \\ \hat{y} \end{array} \quad \begin{array}{c} \hat{x} \\ \rightarrow \\ \text{circle} \\ \leftarrow \\ \hat{y} \end{array} \quad \begin{array}{c} \text{triangle}_{\downarrow 0} \approx \textcircled{0} \\ \text{triangle}_{\downarrow 1} \approx \textcircled{\pi} \end{array} \quad (42)$$

2.2.3 Encoding and Decoding

Diagrammatically, the encoding map is a single spider with one doubled wire as output and one single wire as input. The color of the spider is the basis in which the classical bit gets encoded into. For example, encoding classical information p on a diagonal density matrix ρ into a quantum state in the white basis with indices $i \in \{0, 1\}^n$:

$$\text{triangle}_{\downarrow p} \stackrel{28}{=} \text{triangle}_{\downarrow p} \stackrel{15}{=} \sum_i \text{triangle}_{\downarrow i} \stackrel{28}{=} \sum_i \text{triangle}_{\downarrow i} \stackrel{8}{=} \sum_i \langle p|i \rangle \text{triangle}_{\downarrow i} = \text{triangle}_{\downarrow \rho} \quad (43)$$

The decoding map is the adjoint of the encoding map. Its behavior follows directly from (43).

Phases get lost when decoding quantum states to classical states. We can rely on our knowledge of phase spiders to see how the phase gets eliminated in the process of **decoding** a quantum state with a phase α to classical information.

$$\text{triangle}_{\downarrow p} \stackrel{22}{=} \text{triangle}_{\downarrow p} \stackrel{20}{=} e^{-i\alpha} \text{triangle}_{\downarrow p} e^{i\alpha} \stackrel{21}{=} e^{i(\alpha-\alpha)} \stackrel{21}{=} \text{triangle}_{\downarrow p} = \text{triangle}_{\downarrow p} \quad (44)$$

2.2.4 Complementary spiders

Alice encoding a classical state into a qubit in a certain basis and then Bob measuring that qubit in a complementary basis results in Bob receiving a random bit:


(45)

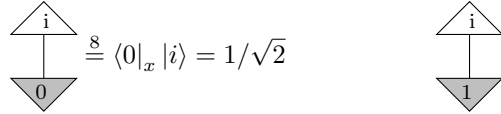
On the contrary, whenever Alice and Bob do respectively encode and measure in the same basis, they achieve successful communication:


(46)

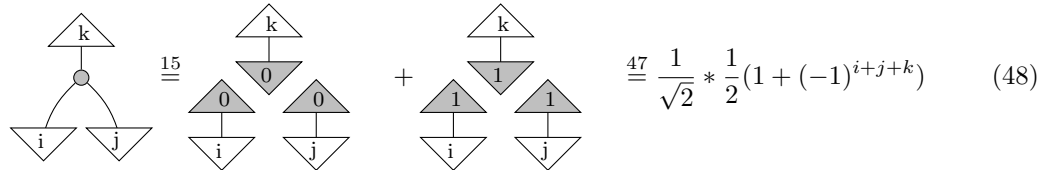
2.2.5 The xor map

The Exclusive OR (xor) map takes two inputs and gives an output of 0 when the inputs are the same and 1 when they are different. Mathematically, if we take $x, y \in \{0, 1\}$ as inputs the output of the xor gate is $(x + y) \bmod 2$. We could just define a diagrammatic map and give it this property. However, there is another way of creating the xor map diagrammatically which only uses spiders. This has the advantage of allowing us to use spider rules to move xor maps around in diagrams.

If we enter two outputs of a certain basis into a spider that is of another basis orthonormal to the inputs this gives the xor map. To see how this works, let's use a gray spider and apply white test in- and outputs. First, recall that:


(47)

Now applying the test in- and outputs to the proposed xor map [3]:


(48)

This equation results in 0 whenever the sum $i + j + k$ is uneven and $1/\sqrt{2}$ when it is even. Now $i + j + k$ is only even whenever $i \oplus j = k$. This thus, up to a factor of $\sqrt{2}$, gives the xor map.

2.2.6 The essential uniqueness of purification

The essential uniqueness of purification is a theorem that states the following: Given two pure maps \hat{V}_1 and \hat{V}_2 that satisfy $V_1, V_2: \mathcal{H} \rightarrow \mathcal{L} \otimes \mathcal{K}$ and:

$$\hat{V}_1 = \hat{V}_2 \quad (49)$$

Then there exists some unitary $U: \mathcal{L} \rightarrow \mathcal{L}$ such that:

$$V_1 = U; V_2 \quad (50)$$

Intuitively, this theorem states that if \hat{V}_1 and \hat{V}_2 are similar to the extent that they satisfy equation (49), then the linear maps V_1 and V_2 can only differ by some unitary U . This is an adaptation of the theorem given by Kissinger and Westerbaan in section 2.1 of [6].

Note that whereas here \hat{V}_1 and \hat{V}_2 have two output wires that are discarded, this theorem scales for maps with an arbitrary number of outputs. In other words, \mathcal{L} could be the Hilbert space of a composite system.

2.2.7 The $k - k'$ commute rule

In this report we will on multiple occasions want to move certain phase spiders past one another. This does, however, require some more advanced diagrammatic derivations that are too extensive for the scope of this report. Therefore we show the result here, and refer to the relevant literature. In [3] on page 771, Coecke and Kissinger introduce the $k - k'$ commute rule and give the full diagrammatic derivation to arrive to this rule. As we will later see, this rule is the main ingredient for pushing Pauli's past one another diagrammatically. It states that for $k, k' \in \{0, \pi\}$:

$$k; k' = k'; k \quad (51)$$

This rule shows us how we can move phase spiders of a different color with either a π or no phase past one another.

3 Protocols

3.1 One Time Pad

The classical One Time Pad (OTP) is perhaps the oldest and most well known cryptographic method. We will first discuss this protocol as a setup to the quantum version. Although originally proposed in 1884 [10], the OTP was first formalized and proven secure in 1949 [13]. Since then, the scheme has essentially remained the same due to its simplicity. A typical OTP goes as follows:

Alice and Bob start with a uniformly random key $k \in \{0, 1\}^n$. Alice wants to send some message $m \in \{0, 1\}^n$ to Bob. She encrypts m using the key k by means of an xor map, generating ciphertext

$x^n = m \oplus k$.¹ After this step, she sends x to Bob who applies k again to x to receive m since $k \oplus x = m$. The assumption on Eve in this scheme is that she doesn't know k and can intercept x . With that being said, if k was truly uniformly random, x is so too, leaving Eve with a random variable even after a successful interception. The main drawback of this scheme is the fact that the key has to be the same size as the message and that this key can only safely be used once.

3.1.1 Diagrammatic Representation

The OTP relies on classical communication channels and xor maps. Since we can represent both of these diagrammatically we can draw out the fully classical OTP protocol by means of diagrams. Recall from section 2.1.5 that the following state generates a random bit:

$$\begin{array}{c} | \\ \circ \end{array} \quad (52)$$

We choose this as the uniformly random key k and give one copy to Alice and one to Bob. Alice, who starts with message m , then applies the xor gate from section 2.2.5 to this message, generating ciphertext x :

$$(53)$$

Note that the dotted line used here is only to illustrate what part of the diagram and Alice "control" and that it bears no physical significance.

Everything Bob now has to do is to apply an xor gate to the ciphertext x and the key k :

$$(54)$$

To check that this is indeed a viable communication protocol we want equation (54) to reduce to:

$$(55)$$

¹In different realizations of the OTP, Alice has different methods of encrypting her message. With that being said, applying an xor to the key and the message is one of the most common versions.

This happens up to a constant factor:

$$(56)$$

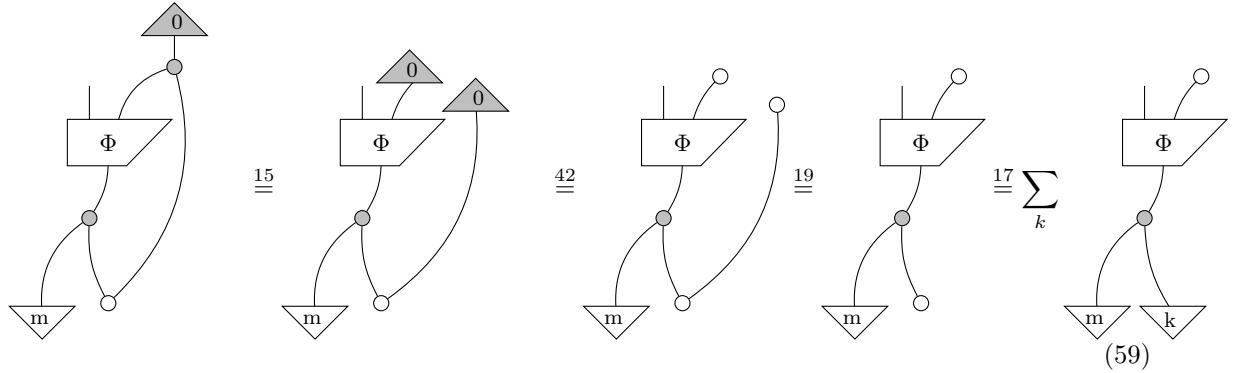
This is a nice result. We see that without any interference from the environment or a third party Alice and Bob achieve uninterrupted communication. But what if there is some eavesdropper that intercepts the communication? We introduce Eve. She receives the ciphertext that is being sent from Alice to Bob, applies some map Φ to it, and subsequently saves one output and sends another to Bob. Note that since the key k is a shared secret between Alice and Bob she doesn't learn about it. Diagrammatically, this situation is as follows:

$$(57)$$

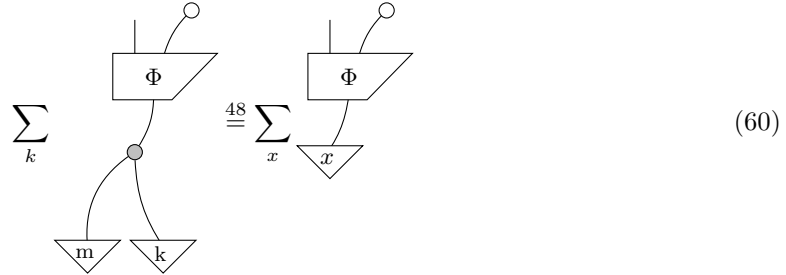
Of course, we want Eve to learn nothing about the message m . To see the situation from her perspective we can discard Bob's output:

$$(58)$$

Using the translations between triangles and spiders from equation (42) and some spider fusion (19):



By equation (48) we know that the two triangles that represent information on m and k get xor'd by the gray spider to give x :



What Eve thus learns in this situation is only the ciphertext, x . This was also what we expected beforehand. Furthermore, since to her this x is random, she does not learn anything on m , meaning that the message is secure.

3.2 Quantum One Time Pad

In the Quantum One Time Pad (QOTP) Alice and Bob share a uniformly random key $\alpha \in \{0, 1, 2, 3\}^n$ and want to communicate some quantum state, ρ . Alice first transforms ρ with a random Pauli map, σ_α . After she has prepared this state she sends it to Bob over a quantum channel. He can then apply the Hermitian adjoint of this random Pauli resulting in the original ρ sent by Alice. This step exploits the unitary property of the Pauli maps. Eve, who doesn't know α , can observe the state sent from Alice to Bob. However, from her perspective, this is the fully mixed state. Mathematically, this can be seen as follows:

$$\rho' = \sum_{\alpha=0}^3 \frac{1}{4} \sigma_\alpha \rho \sigma_\alpha^\dagger = \frac{1}{2} \mathbb{I} \quad (61)$$

In principle, the QOTP is similar to the OTP. By means of encrypting a piece of information with a random key which is unknown to Eve but shared by Alice and Bob the latter can achieve secure communication. As a communication protocol, QOTP is not very efficient. Alice and Bob need use two bits as a key per qubit of communicated information.

3.2.1 Diagrammatic Representation

In the diagrammatic formalism we can construct a set of maps that have the same property as the Pauli maps in equation 61, the Bell maps. These are defined in terms of the Pauli matrices as follows [3]:

$$\sigma_0 = B_0 \quad \sigma_1 = B_1 \quad \sigma_2 = iB_3 \quad \sigma_3 = B_2 \quad (62)$$

In the appendix, these Bell matrices are written out. Note that since they have the same properties as Pauli matrices we use them interchangeably throughout the report.

In order to form diagrams for these Bell maps we have to understand them a little bit better. One of their properties is that they can be constructed from rotations on the Bloch sphere around the z and x axes [5], also known as bitflips around these axes. Note that the maps that apply bitflips are themselves actually Pauli maps. It was chosen to give them their own notation such that they more explicitly represent their behavior. X refers to a bitflip around the x axis and Z refers to a bitflip on the z axis. For random $u, w \in \{0, 1\}$ and $\alpha \in \{0, 1, 2, 3\}$ dependent on u and w we can write this mathematically as follows:

$$Z^u X^w = B_{uw} = B_\alpha \text{ (where } \alpha = u + 2w) \quad (63)$$

Where Z and X denote the maps that do a bitflip in the z and x bases respectively. Note that $Z = \sigma_3$ and $X = \sigma_1$.

Let's see if we can construct these smaller constituents, Z^u and X^w , diagrammatically. Starting with X^w , remember that a bitflip in the x basis corresponds to a rotation around the z axis on the Bloch sphere. Therefore we need to find a diagrammatic map that either rotates the Bloch sphere around its z axis by π radians (X^1) or not (X^0). Since a single spider without a phase is the identity and the z basis is represented by the color white diagrammatically, we can use the white phaseless spider for the case where we don't flip the bit or rotate around the z axis. Let's send the 0 state through in the gray (x) basis to test if this map behaves as expected:

$$\begin{array}{c} \text{Diagram 1: A gray spider with a white circle on top, labeled 0.} \\ \text{Diagram 2: A gray spider with a white circle on top, labeled 0.} \\ \text{Diagram 3: A gray spider with a white circle on top, labeled 1.} \\ \text{Diagram 4: A gray spider with a white circle on top, labeled 1.} \\ \text{Diagram 5: A gray spider with a white circle on top, labeled 1.} \\ \text{Diagram 6: A gray spider with a white circle on top, labeled 1.} \\ \text{Diagram 7: A gray spider with a white circle on top, labeled 1.} \\ \text{Diagram 8: A gray spider with a white circle on top, labeled 1.} \\ \text{Diagram 9: A gray spider with a white circle on top, labeled 1.} \\ \text{Diagram 10: A gray spider with a white circle on top, labeled 1.} \end{array} \quad (64)$$

This map does seem to behave appropriately. We see that whenever we send a gray 0 through, we get a gray 0 as output. Following similar logic, we could also use the white spider with a π phase to make a rotation around the z axis, and thus a bitflip in the x basis. Let's test if this works too:

$$\begin{array}{c} \text{Diagram 1: A white spider with a gray circle on top, labeled 0.} \\ \text{Diagram 2: A white spider with a gray circle on top, labeled 0.} \\ \text{Diagram 3: A white spider with a gray circle on top, labeled 1.} \\ \text{Diagram 4: A white spider with a gray circle on top, labeled 1.} \\ \text{Diagram 5: A white spider with a gray circle on top, labeled 1.} \\ \text{Diagram 6: A white spider with a gray circle on top, labeled 1.} \\ \text{Diagram 7: A white spider with a gray circle on top, labeled 1.} \\ \text{Diagram 8: A white spider with a gray circle on top, labeled 1.} \\ \text{Diagram 9: A white spider with a gray circle on top, labeled 1.} \\ \text{Diagram 10: A white spider with a gray circle on top, labeled 1.} \end{array} \quad (65)$$

These maps do exhibit the correct properties. Indeed, whenever we send a gray 1 through we also get the expected results. It seems that we have formed the X^0 and X^1 maps. While we will not go through the full math here, the Z^0 and Z^1 maps follow very similar logic and also behave as expected. Perhaps unsurprisingly, the former is a gray phaseless spider whereas the latter is a gray spider with a phase π .

Now that we have created our four constituent maps we are in principle ready to make any Bell map of choice. However, we do not want to make a choice, we want to select a random Bell matrix. We still need to find a diagrammatic method to generate a random u and w on which our choice of Bell matrices depend. Recall that generating a random bit is the following diagram:

$$\text{discarding} \stackrel{17}{=} \sum_i \text{triangle}_i \quad (66)$$

We could encode this random classical bit into a qubit as follows:

$$\text{qubit} \stackrel{17}{=} \sum_i \text{triangle}_i \stackrel{43}{=} \sum_i \text{triangle}_i \quad (67)$$

The previous two equations are entirely analogous for the gray (x) basis.²

We now have the necessary ingredients to make a map that applies a random bitflip or not in x and another map that makes a random bitflip or not in z . The results are as follows:

$$X^w \Leftrightarrow \text{diagram}_1 \quad Z^u \Leftrightarrow \text{diagram}_2 \quad (68)$$

Where the gray spider in the X^w map determines the random choice of w and the white spider in Z^u determines the random choice of u .

Let's confirm that the diagrammatic version of X^w does indeed behave as expected:

$$\text{diagram}_1 \stackrel{17}{=} \text{triangle}_0 + \text{triangle}_1 \stackrel{43}{=} \text{triangle}_0 + \text{triangle}_1 \stackrel{42}{=} \text{diagram}_2 + \text{diagram}_3 \stackrel{21}{=} \text{diagram}_4 + \text{diagram}_5 \quad (69)$$

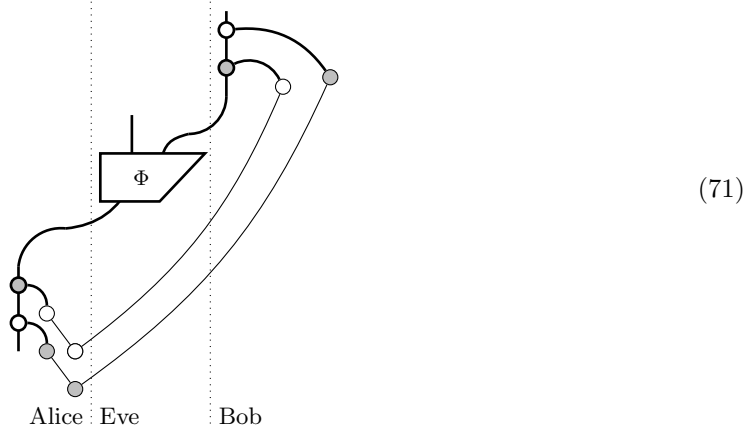
So depending on whether the random bit was a 0 or a 1, we get the map from equation 64 (X^0) or the map from equation 65 (X^1). Again, this derivation is analogous for the map Z^u . We have thus successfully made the constituents to our Bell maps. The random Bell map itself is then as follows:

$$B_\alpha \stackrel{63}{=} Z^u X^w \Leftrightarrow \text{diagram}_6 \quad (70)$$

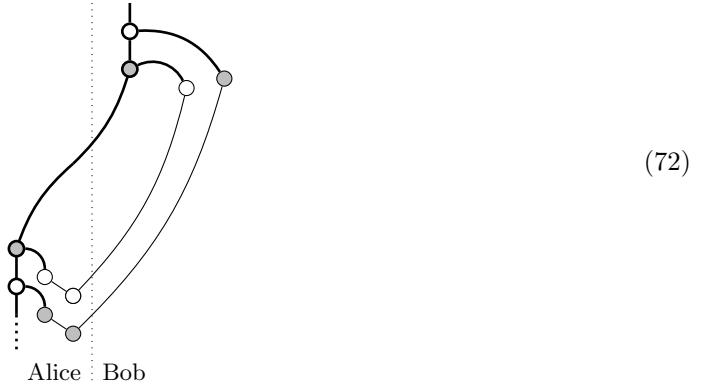
To make a diagrammatic version of QOTP we need to compose the Hermitian adjoint of this Bell map to its input. The bottom Bell map can then be seen as Alice encrypting her state and the

²In fact, we can make an even stronger statement about the diagrams in equation 67: They are actually equal to the analogous diagrams in the gray basis. Note that the rightmost diagram is the adjoint of discarding, which can be interpreted as preparing the fully mixed state, up to a number. Since this state is independent of the basis, these diagrams of white basis are equal to the analogous diagrams in the gray basis.

top Bell map can be seen as Bob decrypting this state. We also need to make sure that Alice and Bob make the same choice in α and thus use the same Bell map. This can be done by connecting the outputs of the spiders that generate the random numbers to the Bell maps of both Alice and Bob. Furthermore, whereas in the previously described protocol a state ρ is communicated as an example, here we can be more general. We allow the state that Alice wants to communicate to Bob to be any state in any basis. This is represented by a dotted line at the bottom of the diagram. Finally, we need to introduce Eve. She goes in between the Pauli's of Alice and Bob. She can be interpreted as an eavesdropper, but also more generally as the environment. In this sense, Eve could thus also just be a source of noise. She gets the map Φ that takes as input the state sent by Alice, and outputs one state that she keeps and another that is sent to Bob. In all, this makes QOTP to be the following diagram:



Let us first consider what would happen if there was no noise, and thus no Eve, in the quantum communication channels. Diagrammatically, this looks as follows:



For this protocol to be a good candidate for communication we need equation (72) to reduce to the following:

$$(76)$$

With D the dimension of the system.

This might give some intuition for what Eve would receive after Alice has encrypted her state. However, in the full protocol, Alice can not just discard her random u and w . Somehow she has to communicate them with Bob. The diagram of QOTP as a communication protocol and Eve with the ability to intercept the quantum state but not u or w is as follows:

$$(77)$$

Where Eve controls the map Φ and receives the rightmost output of this map.

Let's see what happens when we look at this situation from Eve's perspective and thus trace out (discard) Bob's channel.

$$(78)$$

We could now use some of the rules that we have developed so far to propagate the discarding map further down. However, we could also realize that from Eve's perspective there is no difference between Bob applying two Pauli's to his qubit and then discarding it, and Bob discarding his state even before applying these Pauli maps. Therefore, we can write:

$$(78) = \begin{array}{c} \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \\ \vdots \end{array} \begin{array}{c} \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \\ \vdots \end{array} \begin{array}{c} \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \\ \vdots \end{array} \quad (79)$$

The third diagram in equation (78) is up until Eve's map Φ the same as equation (76). We therefore see that from Eve's perspective Alice gives her the fully mixed state due to the Bell map that she applies to her original state. So even in the full protocol, the Bell map is responsible for encrypting the state such that Eve can extract no information from it.

3.3 Quantum Teleportation

In quantum teleportation, Alice and Bob want to communicate a qubit, ρ_1 , using the fact that they share an EPR pair, ρ_{23} , and a classical communication channel. Note that the subscripts in this section help keep track of the different particles present. Subscript 1 refers to Alice's particle, 2 to her side of the EPR pair and 3 to Bob's side of the EPR pair. Alice measures ρ_1 together with ρ_2 in a Bell measurement, entangling these two states to give the total entangled state of ρ_{123} . This projects ρ_3 to one of four possible pure states dependent on the result of the Bell measurement done by Alice. Using the classical communication channel, Alice can send this measurement outcome to Bob in two classical bits. In order for Bob to extract ρ_1 from his part of the entangled state, ρ_3 , he needs to apply to it the Bell map that corresponds to the measurement outcome of Alice's Bell measurement.

The name of this protocol might lead one to think that it includes the teleportation of information. However, this is not the case. This protocol does not achieve faster than light communication. Although the projection of ρ_3 onto a pure state happens simultaneously with Alice making the Bell measurement, in order for Bob to receive any information on ρ_1 from ρ_3 he must apply the Bell map that corresponds to the measurement result of Alice's Bell measurement. Alice thus has to communicate classically to Bob the result of her Bell measurement.

Quantum teleportation carries some similarity to QOTP. In both protocols, Alice and Bob communicate two classical bits of information in order to communicate one qubit. Also, in both cases, Bob applies a Bell map. These similarities might give some intuition for the equivalence of these protocols. In the end of this section we will see this equivalence diagrammatically.

3.3.1 Diagrammatic Representation

The only component of this protocol that we have not worked out diagrammatically yet is the Bell measurement. We know that the Bell measurement is the adjoint of the Bell state. Therefore it suffices to create this Bell state and take its adjoint to make a Bell measurement.

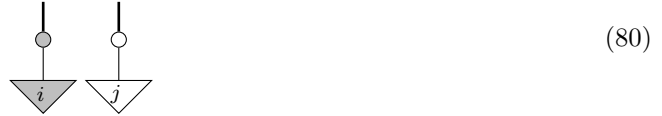
Let's see if we can take the Bell state apart and construct its smaller constituents diagrammatically. The Bell state is actually a composition of a Hadamard (H) and a CNOT gate. The Hadamard gate is applied first to one of the qubits. It transforms $|0\rangle$ to $(|0\rangle + |1\rangle)/\sqrt{2}$ and $|1\rangle$ to $(|0\rangle - |1\rangle)/\sqrt{2}$. These superposition states actually form the basis states in a different basis. Therefore, the Hadamard gate can be seen as a basis transformation. For example, a Hadamard

gate applied to basis states in the z basis will transform those states to the x basis. The CNOT gate is then applied to the two qubits. In the CNOT gate, one input qubit controls whether or not the other is negated. If the former is $|0\rangle$ the latter undergoes the identity and if the former is $|1\rangle$ the latter is negated. The control qubit in the CNOT gate is in this case the output of the Hadamard gate. The result of these operations entangles the two input qubits and puts them in one of four possible Bell states. Let's test this construction on input state $|00\rangle$ to see if it indeed behaves as expected:

1. The Hadamard gate puts the first qubit in superposition: $|00\rangle \Rightarrow \frac{(|0\rangle+|1\rangle)|0\rangle}{\sqrt{2}}$
2. The CNOT negates the second qubit if the first is $|1\rangle$: $\frac{(|0\rangle+|1\rangle)|0\rangle}{\sqrt{2}} \Rightarrow \frac{(|00\rangle+|11\rangle)}{\sqrt{2}}$

The result is thus indeed one of the Bell states, namely B_0 .

Diagrammatically, we do not have to construct the whole Hadamard gate to put one of the qubits into the required superposition. In fact, all we have to do is encode one qubit into a different basis than the other and choose one of the two bases to read the results in. The first part of our diagrammatic version of the Bell state thus becomes:

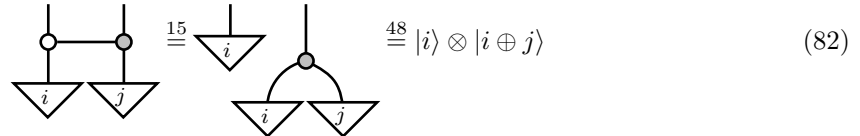


Where i and j are test inputs.

The CNOT gate itself is composed by a copy map and an XOR map. The first qubit is copied after which the second is XOR'd with the copied qubit [12]. Diagrammatically:



Applying test states:



Which is indeed the behavior of the CNOT gate.

Composing equations (80) and (81) then gives the Bell state:



Let's again apply the state $|00\rangle$ to test the functionality of this Bell state preparation map:

$$\begin{aligned}
& \begin{array}{c} \text{Diagram 1} \\ \text{Diagram 2} \\ \text{Diagram 3} \end{array} \stackrel{43}{=} \begin{array}{c} \text{Diagram 4} \\ \text{Diagram 5} \end{array} \stackrel{42}{=} \begin{array}{c} \text{Diagram 6} \\ \text{Diagram 7} \end{array} \stackrel{21}{=} \frac{1}{2} \left(\begin{array}{c} \text{Diagram 8} \\ \text{Diagram 9} \end{array} \right) + \frac{1}{2} \left(\begin{array}{c} \text{Diagram 10} \\ \text{Diagram 11} \end{array} \right) \\
& \stackrel{19}{=} \frac{1}{2} \left(\begin{array}{c} \text{Diagram 12} \\ \text{Diagram 13} \end{array} \right) + \frac{1}{2} \left(\begin{array}{c} \text{Diagram 14} \\ \text{Diagram 15} \end{array} \right) \stackrel{42}{=} \frac{1}{2} \left(\begin{array}{c} \text{Diagram 16} \\ \text{Diagram 17} \end{array} \right) + \frac{1}{2} \left(\begin{array}{c} \text{Diagram 18} \\ \text{Diagram 19} \end{array} \right) \\
& \stackrel{21}{=} \frac{1}{2} \left(\begin{array}{c} \text{Diagram 20} \\ \text{Diagram 21} \end{array} \right) + \frac{1}{2} \left(\begin{array}{c} \text{Diagram 22} \\ \text{Diagram 23} \end{array} \right) \stackrel{42}{=} \frac{1}{2} \left(\begin{array}{c} \text{Diagram 24} \\ \text{Diagram 25} \end{array} \right) + \frac{1}{2} \left(\begin{array}{c} \text{Diagram 26} \\ \text{Diagram 27} \end{array} \right)
\end{aligned} \tag{84}$$

The result is thus indeed the expected Bell state. Note that whereas usually the constants in the Bell states are $1/\sqrt{2}$ here they are $1/2$ since the final state is doubled.

Let's choose the entangled state of 2.1.10 for state ρ_{23} . We can then compose the Bell measurement that Alice applies on the left of this entangled state and the Bell map that Bob applies to the right. Alice also inputs ρ_1 to the Bell measurement. Using a similar approach to the one we took in 3.2.1, we can generalize this to say that this state can in fact be any quantum state. Again, we can represent this by a dotted line which enters the diagram where ρ_1 would go. Finally, we need to make sure to connect the outputs of Alice's Bell measurement to Bob's Bell map. The protocol for teleportation then becomes:



It is redundant to see what happens to Eve or to show that this protocol allows for communication. This is because we can directly tell that this protocol is equivalent to the QOTP protocol described in section 3.2, which was shown to allow for communication without Eve. Placing QOTP without Eve and quantum teleportation side by side, this equivalence is strikingly simple. Everything we have to do is to make a small adjustment to QOTP, writing it a bit more suggestively:

(86)

3.4 Quantum Key Distribution

The first to develop a secure quantum key exchange protocol were Bennett and Brassard, who published an article on QKD in 1984. In order to understand their version of QKD, subsequently referred to as BB84, we first should understand the method of encoding information into quantum states employed in their protocol. This will also help us later on, since this method of encoding has become the standard and will be used in the QKR protocol as well. For the previous two diagrams, we did not care about how the information was encoded into quantum states since we only required Alice and Bob to communicate quantum states. In QKD and QKR, we do include the part where Alice and Bob actually encrypt classical information into a quantum state.

BB84 encoding uses an ubiquitous resource as information carrier, light. The constituents of light, photons, have the convenient property that their polarization forms a two level quantum system. The two orthonormal bases in this quantum system are diagonal (D) and rectilinear (R). In the rectilinear basis we can encode a 0 and a 1 as a photon polarized to 0 and 90 degrees respectively. In the diagonal basis we can do the same for 45 and 135 degrees. This is a valid system of two mutually unbiased bases. Encoding any bit in one basis and then measuring in the other gives a random result. On the other hand, encoding a bit in one basis and measuring in the same gives the correct result. In BB84, Alice uses this method to encode a random bitstring $k \in \{0, 1\}^n$ into a series of qubits to be sent to Bob. She randomly chooses a set of bases to encode them in as well, $b_A \in \{R, D\}^n$. This creates n qubit states with each k_i encoded as a photon of a certain polarization in basis b_i . The set of qubit states, and thus the photons, that this results in is denoted as $|\Psi\rangle = \bigotimes_{i=1}^n |\psi_{k_i}^{b_i}\rangle$. She sends this $|\Psi\rangle$ to Bob who measures in a random basis $b_B \in \{R, D\}^n$. On a public channel, Bob sends b_B back to Alice who compares it with b_A . Alice then tells Bob where he measured in the correct basis, or for what i $b_{A_i} = b_{B_i}$. If there would be no noise in the (quantum) communication channels and no Eavesdropper (Eve) who disturbs the communication Bob should now have the k_i for all the qubits that he measured in the correct basis. Alice and Bob should then have some shared secret key consistent of the bitstring k_i where $b_A^i = b_B^i$. In the case where we do allow Eve to intercept the signal, Alice and Bob should execute one more step to test whether their shared key is really secure. For this step, Alice and Bob can use the fact that any attempt by Eve to measure the communicated quantum states disturbs them. Bob can send part of his result to Alice, who can compare it to the string she originally encoded. If these are the same, it is likely there was no Eve. If they are different, they can be sure that Eve attempted some measurement of the qubits. Finally, if Alice and Bob did achieve successful key exchange, they can safely use their k as future key for One Time Pad encryption (OTP).

3.4.1 Diagrammatic Representation

QKD is the only protocol that has been diagrammatically (somewhat) thoroughly discussed in literature [6]. Therefore, we will not go over the full security proof here. This section is meant to get the reader to understand the diagrammatic version of QKD, such that we can use it later on in the context of equivalences.

To write it out diagrammatically we need to generalize it to a protocol that is not dependent on the physical realization of the qubits. In the new protocol, we therefore say that Alice and Bob have the choice to encode their qubit in any two mutually unbiased bases, which we choose to be the z and x bases. Not considering Eve, if Alice and Bob then measure in the same basis, their quantum state is sent through just fine:


(87)

However, due to the fact that the bases are mutually unbiased we get the following situation when Alice and Bob's measurement bases do not agree:


(88)

These are the cases that Alice and Bob decide to not use upon discussing their choice of bases later in the protocol.

Now if we limit Eve's interference to only a measurement in either the z or x bases she will have a 50% chance of measuring in the correct basis. In the case where Alice and Bob both use the x basis Eve can thus measure correctly:


(89)

Or she measures in the wrong basis and both her and Bob receive a random bit:


(90)

This puts heavy constraints on Eve. Realistically, she could measure in different bases, or find even different ways to extract information from Alice's qubits. Therefore, it is more general to state that Eve can apply some map Φ , on which we can put constraints as needed:


(91)

This is exactly how Kissinger et al. approach this problem in [6]. To prove security for noiseless communication they go on to claim that for Eve's intervention to be undetected whenever Alice and Bob's measurement results are the same, from their perspective Eve's channel must exhibit the following property:

$$\text{Diagram 1} = | \quad \wedge \quad \text{Diagram 2} = | \quad (92)$$

From here on Kissinger et al. in [6] include an extensive (fully diagrammatic) security proof which finally reaches the conclusion that for any Φ that satisfies 92 and mutually unbiased bases x and z we have:

$$\text{Diagram 3} = \text{Diagram 4} = | \quad (93)$$

In words: Eve's channel has no connection to Alice and Bob's and she thus learns nothing about the state Alice sent to Bob.

3.5 Quantum Key Recycling

The part of QKR in which Alice and Bob actually communicate a quantum state is very similar to the previously discussed protocols QKD and QOTP. The main differences between them are in the classical processing part and the encoding of the quantum states. In QKR, Alice and Bob both share knowledge beforehand about some set of bases $b \in \mathcal{B}^n$ and some one-time pad $j \in \{0, 1\}^n$. Before Alice encodes her message $M \in \{0, 1\}^m$ into a quantum state to be sent to Bob, she applies some error-correcting code $M \rightarrow E(M) \in \{0, 1\}^n$ to it and xor's $E(M)$ with j to produce the payload $x \in \{0, 1\}^n$. The error-correcting code allows Bob to determine and correct errors that may arise as a result of a noisy communication channel in the received payload at the cost of added redundancy [?]. The one-time pad is used to mask the message. After the payload is ready Alice prepares $|\Psi\rangle = \bigotimes_{i=1}^n |\psi_{x_i}^{b_i}\rangle$ and sends it to Bob. Bob, since he has the same set of bases should get the correct measurement result, x . Since he also knows j he can xor x and j to receive M , the message from Alice. In the case where there is no Eve detected he can use the same k and b to safely send a new payload back to Alice, who can then send one back to Bob, and so forth. However, if one of the two detects Eve's interference, both parties have to decide on a new key, which they can generate from previously uncompromised transmissions. Eve is allowed to attempt a measurement of the qubits transmitted between Alice and Bob, $|\Psi\rangle$. However, even if she would measure all of the qubits correctly, she only receives x . She would have to xor x with j to receive the message M but she does not know j .

Recently, it was proposed to use eight-state encoding in QKR protocols [4]. This method of encoding quantum states does not rely on the two mutually unbiased bases that we have seen up to this point, it essentially encodes bits into four different non mutually unbiased bases. Allowing each basis two states then results in the characteristic total of eight different states, hence the name of this encryption. The rest of the protocol then proceeds similarly. In the paper in which this was originally brought forward [4], the authors suggest to place each of these states at equidistant locations on the Bloch sphere, corresponding to the eight corners of a cube if it were placed perfectly inside of the Bloch sphere. One of the bases is defined as having a 0 state $|\psi_0\rangle$ at $(1, 1, 1)^T/\sqrt{3}$ and a 1 state $|\psi_1\rangle$ at $(-1, -1, -1)^T/\sqrt{3}$. The others are defined as two Pauli transforms with respect to this basis, giving the following eight states:

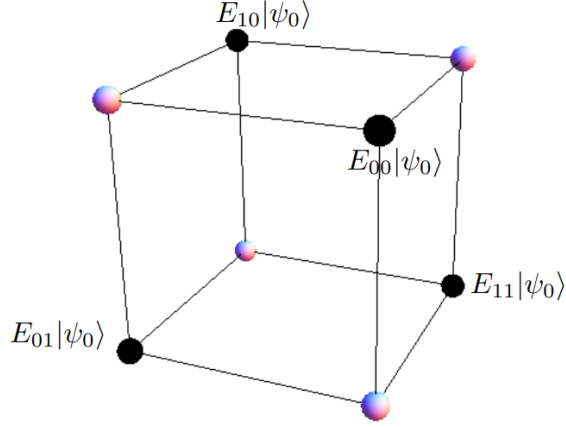


Figure 1: A cube representing the eight cipherstates $E_{uw}|\psi_g\rangle$ as its corner points $(\pm 1, \pm 1, \pm 1)^T/\sqrt{3}$ [4].

$$|\psi_{uwg}\rangle \equiv X^w Z^u |\psi_g\rangle \text{ where } u, w, g \in \{0, 1\} \quad (94)$$

More compactly, we can define $E_{uw} \equiv X^u Z^u$ and have E_{uw} be our encryption operator. This gives rise to the representation of these eight states as given in figure 1.

3.5.1 Practical Implementation

We need to introduce some new diagrammatic notation for eight state encoding. After all, we need to move from a system of two bases to one of four. With that being said, a large part of it should be rather intuitive since we have essentially seen most of it already. The quantum one time pad is actually an eight state encryption protocol. The main difference is that in the protocol introduced for QKR, the eight states are distributed differently over the Bloch sphere. Diagrammatically, we can exploit the similarities by borrowing the notation that was previously introduced for the Pauli's in the QOTP. According to equation (94), we only need to define the properties of one of the bases of eight state encoding and get the others for free by applying Pauli's to the states of this original basis. We choose the color red for this new basis. The red basis has the following two basis states: $E_{00}|\psi_0\rangle = \mathbb{I}|\psi_0\rangle = |\psi_{000}\rangle$ and $E_{00}|\psi_1\rangle = \mathbb{I}|\psi_1\rangle = |\psi_{001}\rangle$. We should first define triangles that represent classical information on these states, similarly to how we have classical triangles for the gray and white bases. We use the color red for this as well. This gives us the following diagrammatic components for the preparation of a state and the effect in this basis:

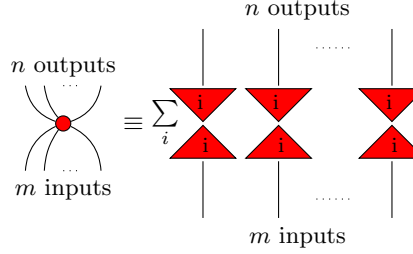
$$|\psi_{00g}\rangle \equiv \begin{array}{c} | \\ \text{red triangle down with 'g' inside} \end{array} \quad \langle\psi_{00g}| \equiv \begin{array}{c} \text{red triangle up with 'g' inside} \\ | \end{array} \quad (95)$$

Note that we have as well that:



$$\equiv \delta_{ij} \quad (96)$$

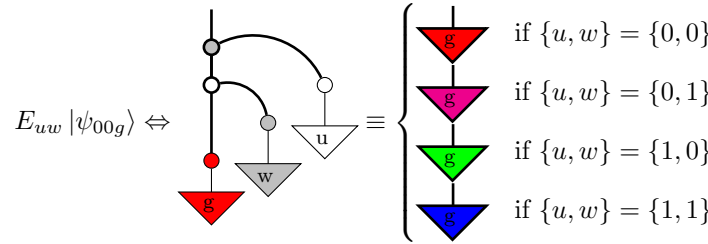
Finally, we need to define a spider that encodes these $|\psi_{000}\rangle$ and $|\psi_{001}\rangle$ to the respective quantum states. In fact, since equation (15) is independent for the choice of basis, we can just copy it and exchange the white triangles for red ones:



$$\equiv \sum_i \quad (97)$$

By a combination of equations (96) and (97) it is trivial to check that the red spider follows the spider rules as they are given in section 2.1.5. On the other hand, we have to tread carefully regarding its interaction with differently colored spiders, since we cannot exploit mutual unbiasedness.

We are now in principle ready to employ eight state encoding as it was proposed in [4] diagrammatically. Equation (70) tells us how we can make Pauli's diagrammatically, and equation (94) tells us that we can create all of our cipherstates by applying these Pauli's to the recently defined new basis states $|\psi_{00g}\rangle$. Defining a new color for each of the bases, we could thus encode all of our eight different cipherstates as follows:



$$\equiv \begin{cases} \text{red spider} & \text{if } \{u, w\} = \{0, 0\} \\ \text{pink spider} & \text{if } \{u, w\} = \{0, 1\} \\ \text{green spider} & \text{if } \{u, w\} = \{1, 0\} \\ \text{blue spider} & \text{if } \{u, w\} = \{1, 1\} \end{cases} \quad (98)$$

Since the different bases will reappear quite often, we can save ourselves from unnecessarily complex diagrams by redefining each combination of u and w into a new spider, each with a different color, corresponding to the colors chosen for the different bases in equation (98). These eight state encoding spiders represent the four bases of eight state encoding in much the same way the gray and white spider represent the two bases of four state encoding.

$$\{u, w\} = \{0, 0\} \Leftrightarrow \bullet \quad \{u, w\} = \{0, 1\} \Leftrightarrow \bullet \quad \{u, w\} = \{1, 0\} \Leftrightarrow \bullet \quad \{u, w\} = \{1, 1\} \Leftrightarrow \bullet \quad (99)$$

To make our lives easier, we introduce one more spider. This spider is actually nothing new, it is the set of the four spiders introduced in 99. Having this spider may come in handy whenever

we want to make a statement that counts for all four of the colored spiders. Instead of making the same statement for each of the different spiders, we can then make it once for this new spider. We choose the color brown for this:

$$\text{brown spider} \equiv \{ \text{red}, \text{green}, \text{pink}, \text{blue} \} \quad (100)$$

Finally, we can do the same for triangles of all colors. A brown triangle thus represents the set of all colors of triangles. Using these new tools we can reduce equation (98) to:

$$(98) = \text{brown triangle} \quad (101)$$

Let's first see what our QKR protocol looks like in this new notation:



Alice and Bob respectively encode and measure in the same basis. Eve controls some map Φ and receives one of its outputs. With this all set up we are in principle ready to develop a security proof for noiseless QKR, purely diagrammatically. Our security proof is very similar to the one proposed by Kissinger and Westerbaan in literature [6], where they prove security of four state QKD relying largely on the mutual unbiasedness of the gray and white bases. Although we do bring this mutual unbiasedness indirectly into our security proof as well by incorporating the Pauli's consistent of the gray and white spiders that we have seen before, we aim to extend this proof to also incorporate our four new non-mutually unbiased bases. As we will see, we will rely on gray and white spiders only through the use of these Pauli's, which are a valid member of eight state encoding.

Putting in the property that the quantum channels and maps exhibit no noise according to Alice and Bob is equivalent to saying that Eve goes undetected whenever Alice and Bob measure in the same basis. Since Eve represents the environment this can also be interpreted as the quantum channels being noiseless. We have already briefly discussed the security proof of Kissinger and Westerbaan in section 3.4, where we made the same statement for Alice and Bob using the gray and white bases. Noting that brown represents the set of four colors introduced in equation (100), we can implement this statement for our four new bases as follows:

$$\begin{array}{c} \text{red spider} \\ \text{box } \Phi \\ \text{red spider} \end{array} = \mid \wedge \begin{array}{c} \text{green spider} \\ \text{box } \Phi \\ \text{green spider} \end{array} = \mid \wedge \begin{array}{c} \text{pink spider} \\ \text{box } \Phi \\ \text{pink spider} \end{array} = \mid \wedge \begin{array}{c} \text{blue spider} \\ \text{box } \Phi \\ \text{blue spider} \end{array} = \mid \Leftrightarrow \begin{array}{c} \text{brown spider} \\ \text{box } \Phi \\ \text{brown spider} \end{array} = \mid \quad (103)$$

With this as a starting point we can pre- and postcompose measurement maps as follows:

$$\begin{array}{c} \text{brown spider} \\ \text{box } \Phi \\ \text{brown spider} \end{array} = \text{brown spider} \quad (104)$$

Now we can purify the brown spiders in equation (104) by doubling and discarding an extra output according to equation (34). Note that we can also, without loss of generality, state that $\Phi = \hat{V}$ since Φ already fits the definition of a purified state given in section 2.1.9. This gives rise to the following diagrams:

(105)

Where $\hat{\psi}$ can be any normalized pure state.

By applying the essential uniqueness of purification from section 2.2.6 to equation 105 we receive equation (106). Note that since we discarded multiple outputs in equation (105) we also have to apply the unitary U over multiple outputs:

(106)

Deleting the second and fourth output of the LHS of equation (106) gives the following:

(107)

Doing the same to the RHS of equation (106):

(108)

Note that this holds for spiders of all colors of equation (103). The following is thus also true for spiders of all these colors:

$$(109)$$

We can use this equation (109) to show that V separates. Since V is just the undoubled version of Φ this also implies that Φ separates, which is what we wanted to achieve. Starting with V itself:

$$(110)$$

The strategy is to add something separated from the diagram in equation (110) and put V in there. We can just compose the creation and deletion of a random variable. Adding it to the diagram and writing one output of V a bit more suggestively:

$$(111) = \frac{1}{D}$$

$$(111)$$

Now, we need some method to attach the separated part to the diagram temporarily such that we can move V through. It turns out that eight state encoding gives us a map that can do exactly this: the random Pauli's. Equation (76) already showed how a random Pauli can separate two parts of a diagram. We place it between the two diagrams in equation (111):

$$(111) \stackrel{(76)}{=} \stackrel{(109)}{=} (112)$$

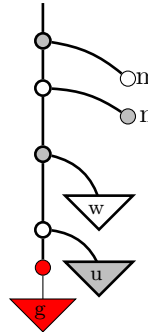
What we need now is some way to pass V through these Pauli, encoding, and decoding maps. Let's have a closer look at just the encoding map and the subsequent random Pauli by applying some test state g to it:


(113)

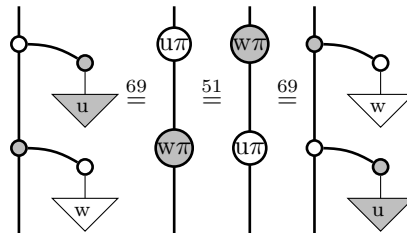
So what we have here is a random Pauli, precomposed by any of the four encoding maps and a test state of the same basis. Since each of the four bases are just Pauli's with respect to one another, combining the encoding and random Pauli maps actually selects one of the four bases to encode in at random. More explicitly, remember by equation (94) that each of the cipherstates can be created by two Pauli's and the $|\psi_g\rangle$ basis states ($|\psi_{uwg}\rangle = X^w Z^u |\psi_g\rangle$). Then applying a random Pauli ($X^n Z^m$ where n and m are random) to any of the cipherstates gives the following:

$$X^n Z^m X^u Z^w |\psi_g\rangle = X^l Z^p |\psi_g\rangle \stackrel{94}{=} |\psi_{lp g}\rangle \quad (\text{where } l \equiv n \oplus u \text{ and } p \equiv m \oplus w) \quad (114)$$

Which, by randomness of l and p , is a cipherstate encoded into a random one of the bases. Encoding a state into a cipherstate with a certain basis and then applying a random Pauli is thus equivalent to encoding this cipherstate into a random basis. We can find this diagrammatically too. Remember that equation (98) tells us how we can encode in any basis by applying two Pauli's with respect to a state in the red basis diagrammatically. The brown spider and state from equation (113) can thus be replaced by a red spider and state, followed by two Pauli's. Looking just at this encoding part and the random Pauli's that follow immediately after we thus get:


(115)

The only ingredient that we are missing is some way to move Pauli's past one another. Mathematically, the fact that we can do this is evident from the anticommutative property of Pauli's. Anticommutativity implies that any two Pauli's σ_i and σ_j satisfy $\sigma_i \sigma_j = -\sigma_j \sigma_i$. Diagrammatically we can commute Pauli's using the $k - k'$ commute rule from section 2.2.7 as follows:


(116)

Putting this back into equation 115 and moving around some spiders gives:

$$\begin{aligned}
(115) &\stackrel{15}{=} \sum_{nm} \text{Diagram 1} \stackrel{116}{=} \sum_{nm} \text{Diagram 2} \stackrel{19}{=} \sum_{nm} \text{Diagram 3} \stackrel{19}{=} \sum_{nm} \text{Diagram 4} \\
&\hspace{15cm} (117)
\end{aligned}$$

By equation (48) we know that the m and w here are xor'd by the gray spider and the n and u are xor'd by the white spider. We define $m \oplus w \equiv p$ and $n \oplus u \equiv l$, where randomness of m and n implies randomness of p and l . This gives:

$$(116) = \sum_{lp} \text{Diagram 5} \hspace{15cm} (118)$$

Which is a cipherstate g encoded into a random basis by equation (98):

$$(118) \stackrel{(98)}{=} \frac{1}{4} \text{Diagram 6a} + \frac{1}{4} \text{Diagram 6b} + \frac{1}{4} \text{Diagram 6c} + \frac{1}{4} \text{Diagram 6d} \hspace{15cm} (119)$$

And thus we see that the encoding and subsequent Pauli's in equation (113) actually encode some input state g into a random one of the four bases. Also note that since g was a brown triangle, this result is independent of the basis of the input state. Encoding into a certain basis and then applying a random Pauli thus has the same effect as applying a random one of the basis spiders. We can put this result into the relevant parts of equation (112) as follows:

$$\text{Diagram 7} \stackrel{(119)}{=} \frac{1}{4} \text{Diagram 8a} + \frac{1}{4} \text{Diagram 8b} + \frac{1}{4} \text{Diagram 8c} + \frac{1}{4} \text{Diagram 8d} \hspace{15cm} (120)$$

By undoubling the wire and using equation (109) we can then move V through all of these:

This is a strikingly simple equivalence. Bending the U-shaped curve in the quantum teleportation protocol such that it becomes a straight line already does the trick. In a sense, quantum teleportation could thus be seen as the EPR variant of the QOTP.

Due to their equivalence, whatever we can prove with the one diagram we can state as fact for the other without needing to prove it. This can be very convenient in security proofs.

4.2 QKD Prepare and Measure and QKD EPR

4.3 Quantum Alice and Silent Bob Equivalences

In a recent article [7], Skorič and Leermakers propose a scheme for QKR that includes no classical communication from Alice to Bob and only one bit from Bob to Alice. To prove its security, the authors modify the proposed protocol to one that is better suited for this in a series of steps that preserve security-wise equivalence to the original. In this section, we will have a look at these equivalences diagrammatically using the eight state encoding QKR protocol from equation (125) as starting point:

4.3.1 Masking the qubit payload with public randomness

In the first equivalent protocol Alice and Bob both apply the same bitstring $a \in \{0,1\}^n$ in the classical domain, before encoding and after decoding respectively. a is public, implying that Eve also learns it. The rest of the protocol remains the same. To implement this diagrammatically, we give Eve another map with respect to (91) to represent her doing classical post processing. In this map, she uses her knowledge on a and what she learned from intercepting the quantum state from Alice to Bob.

We need to create one more set of maps before we can implement this diagrammatically. Since we haven't fully worked out all of the properties of the colored spiders yet, it is more convenient to not use them whenever the protocol doesn't tell us to. Therefore, we propose a map that translates between the white and colored bases. This allows us to do all of the classical processing in terms of white and gray spiders which is necessary if we want to use for example mutual unbiasedness

(2.2.4) or the xor map (2.2.5). The map that translates between the white and red bases is the following:

$$\begin{array}{c} \text{red box} \\ \downarrow \\ \text{red box} \end{array} : |g\rangle_z \mapsto |\psi_{00g}\rangle \quad \begin{array}{c} \text{red box} \\ \downarrow \\ \text{red box} \end{array} : |\psi_{00g}\rangle \mapsto |g\rangle_z \quad (126)$$

This map is worked out in Dirac notation in appendix ???. The other three colors then also have the respective boxes:

$$\begin{array}{c} \text{green box} \\ \downarrow \\ \text{green box} \end{array} : |g\rangle_z \mapsto |\psi_{10g}\rangle \quad \begin{array}{c} \text{green box} \\ \downarrow \\ \text{green box} \end{array} : |\psi_{10g}\rangle \mapsto |g\rangle_z \quad (127)$$

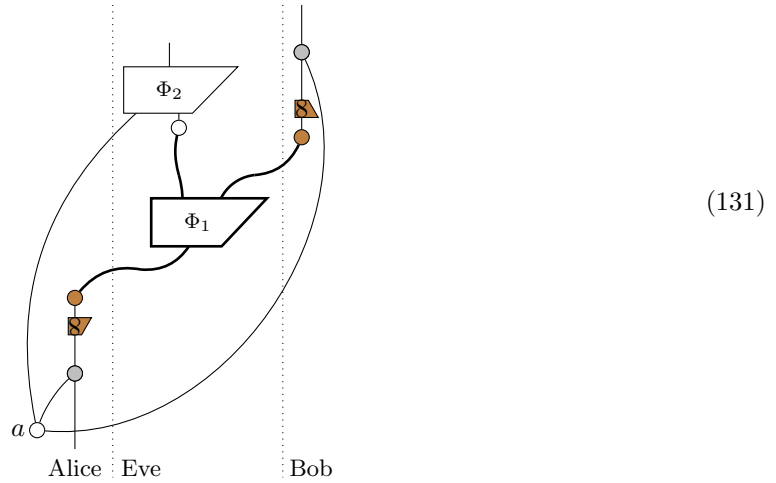
$$\begin{array}{c} \text{pink box} \\ \downarrow \\ \text{pink box} \end{array} : |g\rangle_z \mapsto |\psi_{01g}\rangle \quad \begin{array}{c} \text{pink box} \\ \downarrow \\ \text{pink box} \end{array} : |\psi_{01g}\rangle \mapsto |g\rangle_z \quad (128)$$

$$\begin{array}{c} \text{blue box} \\ \downarrow \\ \text{blue box} \end{array} : |g\rangle_z \mapsto |\psi_{11g}\rangle \quad \begin{array}{c} \text{blue box} \\ \downarrow \\ \text{blue box} \end{array} : |\psi_{11g}\rangle \mapsto |g\rangle_z \quad (129)$$

We call these maps the eight state encryption maps. Again, the brown box is the set of the four individual maps:

$$\begin{array}{c} \text{brown box} \\ \downarrow \\ \text{brown box} \end{array} \equiv \left\{ \begin{array}{c} \text{red box} \\ \downarrow \\ \text{red box} \end{array}, \begin{array}{c} \text{green box} \\ \downarrow \\ \text{green box} \end{array}, \begin{array}{c} \text{pink box} \\ \downarrow \\ \text{pink box} \end{array}, \begin{array}{c} \text{blue box} \\ \downarrow \\ \text{blue box} \end{array} \right\} \quad \begin{array}{c} \text{brown box} \\ \downarrow \\ \text{brown box} \end{array} \equiv \left\{ \begin{array}{c} \text{red box} \\ \downarrow \\ \text{red box} \end{array}, \begin{array}{c} \text{green box} \\ \downarrow \\ \text{green box} \end{array}, \begin{array}{c} \text{pink box} \\ \downarrow \\ \text{pink box} \end{array}, \begin{array}{c} \text{blue box} \\ \downarrow \\ \text{blue box} \end{array} \right\} \quad (130)$$

Including this map, and the fact that Alice and Bob both xor with a public variable a in their classical domain, we have the following diagram:



With Φ_1 the map that Eve uses to intercept the communication and Φ_2 the map that Eve uses to do classical processing on both this intercepted data and a .

For a security proof of this protocol, we can state that Eve's operations introduce no noise from Alice and Bob's perspective whenever they measure in the same basis. This is the same statement that we made in section 3.5 to prove the security of QKR, and in section 3.4 to prove the security of QKD. For the diagram in equation (131), this looks as follows:

$$(132)$$

By causality we then know that applying Eve's map Φ_2 and then deleting the output is essentially the same as not having applied Eve's map altogether:

$$(133)$$

Then we can pre and postcompose another xor with a on both sides of equation (133):

$$(134)$$

According to the diagrammatic rules for mutual unbiasedness given in equation (45) this separates a on both sides of equation (133):

$$(135)$$

Which is equivalent to the same diagrams without a random a :

$$(136)$$

Finally, postcomposing a brown eight state encryption map and precomposing its adjoint and exploiting the fact that it is unitary:

$$(137)$$

This equation we have already seen, it is the starting point for the security proof of eight state encoding QKR, equation (104).³ Therefore, we could just follow all the steps which we followed in

³There is of course a minor difference: In (104) Eve has the map Φ and here Eve has the map Φ_1 . This is insignificant however. Φ_1 was only introduced to distinguish between Eve's quantum map and Eve's classical map. We can just say that $\Phi = \Phi_1$ without loss of generality since Eve's quantum map can be any (multitude of) unspecified operation(s) in both cases.

section 3.5 and get as a result that Φ_1 separates as follows:⁴

(138)

Putting this back into equation (131) we get the following:

(139)

In a series of steps following rules which we have seen before we can show that Eve's map again separates entirely from Alice and Bob's part of the diagram:

(140)

⁴This is also why they are security-wise equivalent, they follow the same security proof. If it turns out that the security proof in section 3.5 for the original QKR protocol only allows for security up to some factor, the protocol with public randomness from this section is also only secure up to that same factor.

So we see that even though Eve gets a and some quantum state, she is not attached to Alice and Bob's communication channel. Alice and Bob on the other hand achieve uninterrupted communication. This is what we would intuitively expect for the noiseless case.

4.3.2 EPR version of the protocol

The second equivalent protocol is an EPR version of the original. We have already seen a diagrammatic implementation of an EPR protocol in section 3.3, the diagram for quantum teleportation. The characteristic difference between an EPR version of a protocol and a normal version is that in the former Alice and Bob share some entangled qubit. In quantum teleportation this is the Bell state, given by the doubled wire shaped like a cup in equation (85).

Diagrammatically, it is rather trivial to go from the protocol introduced in the previous section (131) to an EPR version. In fact, we only need to bend around some wires. We split this up in a few steps to show where all the parts go. First of all, bending around Alice's wire according to the yanking equation (38):

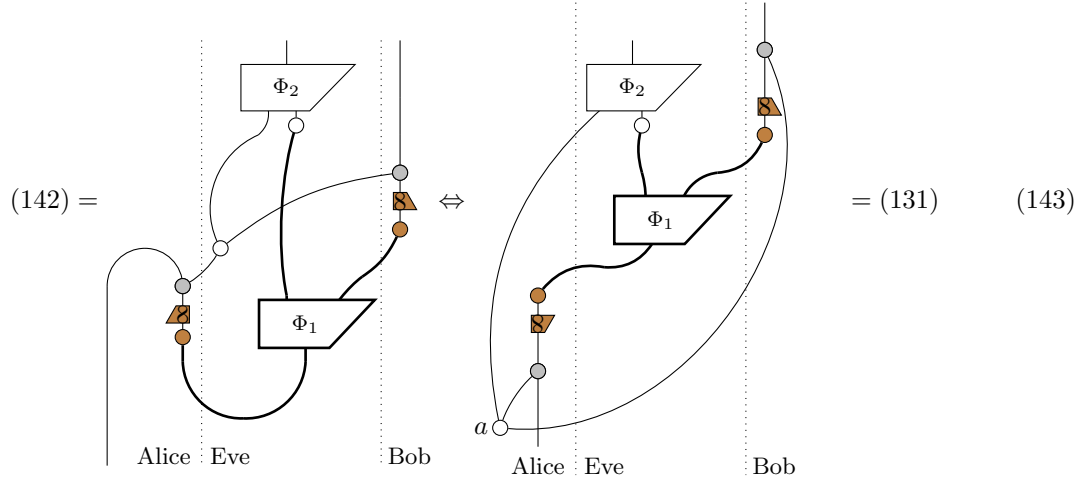
$$(141) = \text{Diagrammatic transformation using equation (38)} \quad (141)$$

And subsequently pulling her encoding and encryption operations through:

$$(141) = \text{Diagrammatic transformation} = \text{Diagrammatic transformation} \quad (142)$$

Alice Eve Bob

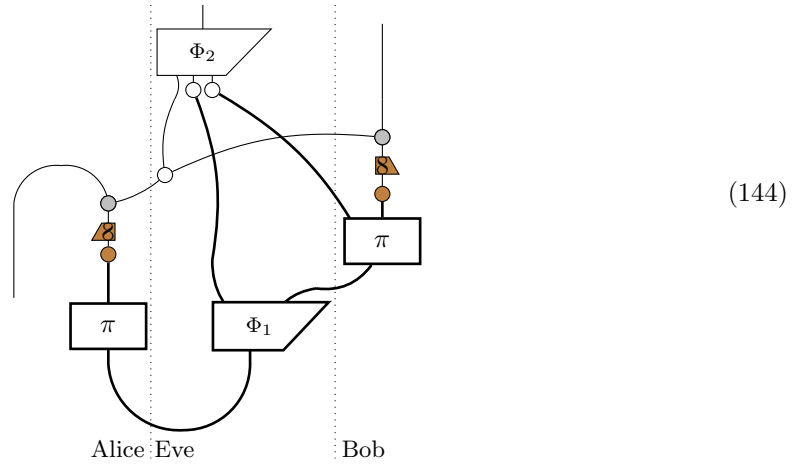
Which is indeed the EPR version of the protocol in equation (131):



Of course, if these protocols are equivalent, then by security-wise equivalence of (131) and (125) we also have security-wise equivalence of the EPR version introduced in this section, (141), and (125), the protocol that we started with in this section.

4.3.3 Adding a random permutation

In the next equivalent step, the authors propose that Alice and Bob both publicly decide on the same random permutation (π) to apply to their own set of qubit states before measuring, and forget it right after. Eve, however, can remember this π . Diagrammatically, this looks as follows:



Intuitively, it is evident that this step preserves equivalence to the original from Alice and Bob's perspective due to the fact that this step preserves entanglement of the EPR states. Whether this protocol works or not for Alice and Bob is independent on the choice of states that they receive, as long as they are entangled. The states that Eve sends to Alice and Bob are entangled up to a certain degree dependent on the amount of noise Eve introduces, and the π 's completely preserve

entanglement since the same operation is applied on both qubits. To make a diagrammatic proof from this intuition we have to first exploit the fact that we can turn Eve's map into a state. We can do this as follows:

(145)

Note that the dotted map in this diagram has no diagrammatic significance and is only intended for didactic purposes

We can show that this is a valid operation diagrammatically. Isolating just the state of Eve in the RHS of equation (145) and realizing that Eve has the freedom to do whatever she wants in her state Φ_1 :

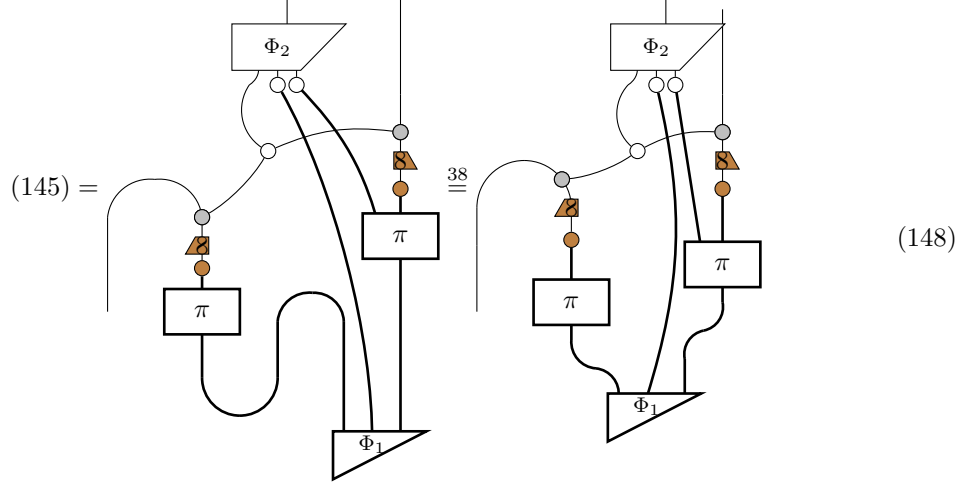
(146)

Then using the yanking equation (38):

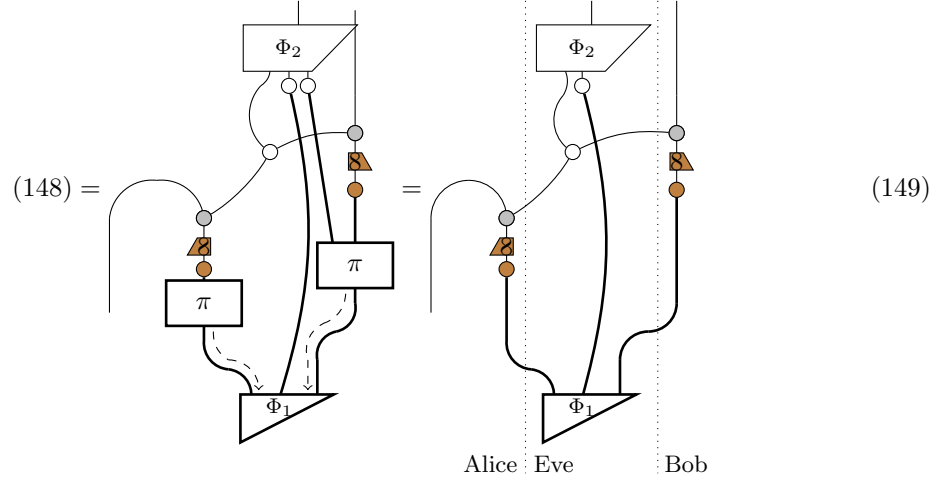
(147)

We see that we can transform Eve's map into a state and reversedly. The operation from equation (145) is therefore valid.

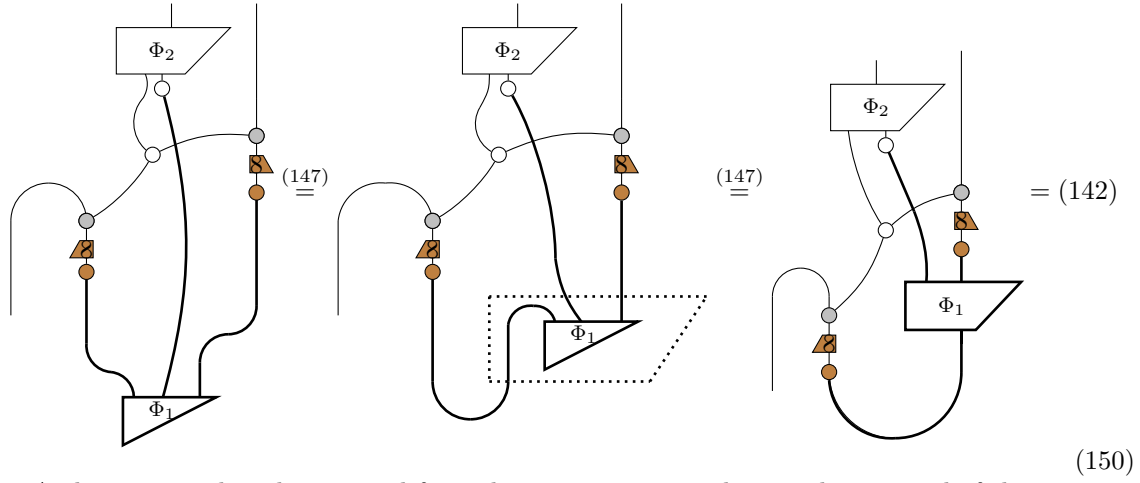
Starting with the RHS of equation (145) and using the yanking equation (38):



Now we are ready to put in the intuition that we had in the beginning of this section. From Alice and Bob's perspective, everything they need is for Eve to prepare and send them an (imperfect) EPR state. If Eve does this in her map Φ_1 the two subsequent random permutations will potentially change the particular choice of EPR states that Alice and Bob receive, but they will remain entangled since the same operation is applied on both sides. Furthermore, Alice and Bob forget π right after they apply it. Therefore, after it is applied, they are indifferent for as to whether they did it or Eve did it.



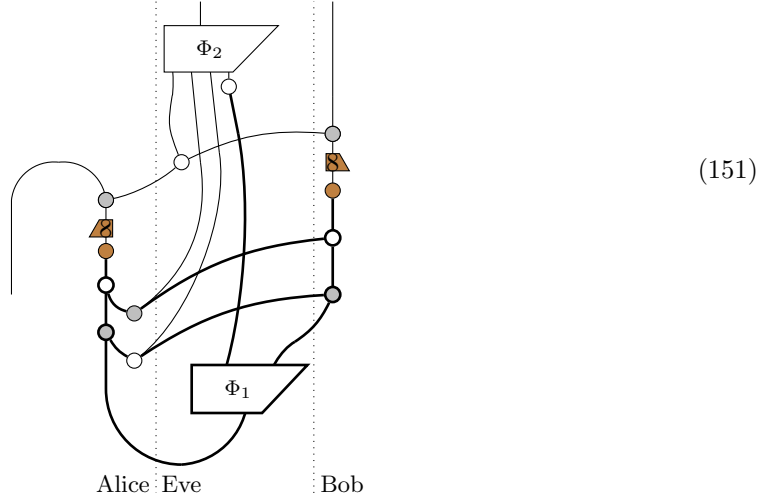
To show that this protocol is equivalent to the previous protocols we can just turn Eve's state back into a map:



And so we see that the protocol from this section is equivalent to the protocol of the previous section.

4.3.4 Adding random Pauli transforms

In the final equivalent step, the authors propose that Alice and Bob both publicly decide on the same random Pauli to apply to their own qubit states before measuring. Afterwards they again forget about their particular choice of Pauli. Diagrammatically, this looks as follows:



The difference between this protocol and the EPR protocol introduced in section 4.3.2 are the two single spiders in the middle which represent the classical random variables that determine the Pauli's which are applied to both Alice and Bob's states. Information on the particular choice of Pauli's also goes to Eve by the thin lines originating from these two single spiders. Let's isolate the Pauli that Alice gets and her measurement and eight state encryption maps:

$$(152)$$

We do not consider the two triangles that go to Bob for now. These simply determine the Pauli's that he applies to his qubit state. All we need to know is that they are the same as Alice's. Furthermore we can take the brown decoding spider of Alice further apart according to equation (98). This equation shows how all the cipherstates can be encoded by a red spider and two Pauli's. Since decoding is the adjoint of encoding, we can decode in any of the four bases by first applying two Pauli's followed by a red spider. To then return back to the classical z basis we need to also exchange the brown eight state encryption box for a red one:

$$(153)$$

This equation is very similar to equation (115). In fact, for equation (115) we showed that encoding in a certain basis and then applying a random Pauli results in encoding in a random basis. Here we will show that applying a random Pauli and then decoding in a certain basis results in decoding in a random basis:

$$\begin{array}{ccc}
 \begin{array}{c} \text{Diagram 1: A vertical line with a red triangle at the top, a red dot, a white circle, and two gray circles. The white circle is connected to a gray triangle labeled 'u'. The first gray circle is connected to a gray triangle labeled 'n'. The second gray circle is connected to a gray triangle labeled 'w'. The line ends in a gray triangle labeled 'm'. To the right are two gray triangles labeled 'n' and 'm' with dotted lines above them.} \\
 (153) \stackrel{116}{=} \sum_{nm}
 \end{array}
 &
 \begin{array}{c} \text{Diagram 2: A vertical line with a red triangle at the top, a red dot, a white circle, and a gray circle. The white circle is connected to two gray triangles labeled 'n' and 'u'. The gray circle is connected to two gray triangles labeled 'm' and 'w'. To the right are two gray triangles labeled 'n' and 'm' with dotted lines above them.} \\
 \stackrel{19}{=} \sum_{nm}
 \end{array}
 &
 \begin{array}{c} \text{Diagram 3: A vertical line with a red triangle at the top, a red dot, a white circle, and a gray circle. The white circle is connected to a gray triangle labeled 'n' and a white circle. The gray circle is connected to a gray triangle labeled 'm' and a white circle. The white circles are connected to gray triangles labeled 'u' and 'w'. To the right are two gray triangles labeled 'n' and 'm' with dotted lines above them.} \\
 \stackrel{19}{=} \sum_{nm}
 \end{array}
 \end{array}
 \tag{154}$$

By equation (48) we know that the gray n and u are xor'd by the white spider and that the white m and w are xor'd by the gray spider. Defining $m \oplus w \equiv p$ and $n \oplus u \equiv l$ and realizing that l and p are random then gives:

$$\begin{array}{ccc}
 \begin{array}{c} \text{Diagram 4: A vertical line with a red triangle at the top, a red dot, a white circle, and a gray circle. The white circle is connected to a gray triangle labeled 'p'. The gray circle is connected to a gray triangle labeled 'l'. To the right are two gray triangles labeled 'n' and 'm' with dotted lines above them.} \\
 (154) = \sum_{nmlp}
 \end{array}
 &
 \begin{array}{c} \text{Diagram 5: A vertical line with a red triangle at the top, a red dot, a white circle, and a gray circle. The white circle is connected to a gray triangle labeled 'p'. The gray circle is connected to a gray triangle labeled 'l'. To the right are two gray circles with dotted lines above them.} \\
 \stackrel{17}{=} \sum_{lp}
 \end{array}
 \end{array}
 \tag{155}$$

As expected, Eve receives the random variables n and m , but she does not learn l and p since these require knowledge of u and w . Alice and Bob thus measure in a random basis and Eve doesn't know this basis. More importantly for Alice and Bob, they measure in the same random basis since for Bob's side, the whole derivation follows the same series of steps. Putting this back into the larger picture, equation (151):

$$\begin{aligned}
(151) &\stackrel{155}{=} \sum_{lp} \text{Diagram 1} = \frac{1}{4} \text{Diagram 2} \\
&+ \frac{1}{4} \text{Diagram 3} + \frac{1}{4} \text{Diagram 4} + \frac{1}{4} \text{Diagram 5}
\end{aligned}
\tag{156}$$

Each of these protocols occur with probability $\frac{1}{4}$. We also know that they are all security-wise equivalent to the protocol in equation (142) since they form the members of the set of protocols in the RHS of equation 142. The protocol in equation (151) thus preserves security-wise equivalence to the protocol that was the starting point of this section, equation (125).

5 Discussion

After reading up to this point, the novelty of this research may not be self-evident. After all, we did not develop any new protocols or gave higher security bounds on existing protocols. Be that as it may, these are not the kind of things that should be expected of the diagrammatic method as of yet. The novelty of this work comes from the fact that it provides diagrammatic realizations of protocols and security proofs that have never been worked out in this fashion before. We took preexisting work and gave it a place in the context of the diagrammatic method for quantum cryptography. As of now it is not yet the task of the diagrammatic method to develop novel protocols or provide higher security bounds. Before we could do any of that, we should first see whether and how we can work up to the current state of the art. Only when we reach that point, we are ready to use this method to actually benefit the field of quantum cryptography itself. This work provides a foundation to reach this point. Kissinger and Westerbaan built this foundation for quantum key

distribution in [6] and in this work we did the same for quantum key recycling with eight state encoding.

With that being said, maybe it is not even the role of the diagrammatic method to eventually provide new protocols or bounds on security proofs. Coecke and Kissinger - in their first book on this method - propose it to be fully independent as an alternative to Dirac notation. Their book is aimed at giving a first introduction into quantum mechanics, where new topics are introduced along with the notation and no knowledge of Dirac notation is assumed. Since this works well, the notation allows for a satisfactory representation of the introductory concepts of quantum mechanics. The question then arises of whether it also allows for the representation of more advanced quantum mechanical topics. In this report, we take it a bit further than in the book of Coecke and Kissinger. We use the diagrammatic method to make security proofs and analyze some recent papers' equivalences. We do at no point, however, assume that Eve introduces noise since whenever we would do this, we would introduce statistics for which we have not worked out the notation yet. Kissinger and Westerbaan in [6] do not shy away from combining the notation with statistics in their security proof for QKD with noise. With that being said, their approach is a bit clumsy since they use equations rather than diagrams to actually work out the statistics. Their security proof by itself is also not state of the art, and is mostly relevant since it is the first security proof for QKD that is conducted mostly with the diagrammatic notation. The position that this method is in right now is that it allows for noiseless security proofs, is useful in visualizing equivalences, and is somewhat capable of security proofs with noise and the statistics that come with it. This is by itself a significant role for this method. Furthermore, due to the fact that it is relatively new - Coecke and Kissinger released their book in 2017 [3] - it may in the future take up more responsibility. As of now it is simply too early to predict what its role will eventually be.

This work leaves open plenty of opportunities for future research. As a direct follow up to this report, a noisy version of the quantum key recycling security proof of this report could be developed. It is possible that a security proof along the lines of the one for noisy quantum key distribution given in [6] also applies to the QKR protocol from section 3.5. Furthermore, since the diagrammatic method is very young and literature on it is scarce, it would benefit from more protocols and their security proofs being written out in this fashion.

6 Conclusion

The aim of this research was largely divided up into two parts. The first was to introduce the reader to the diagrammatic method. Therefore, chapter 2 functions as a sort of crash course to the diagrammatic notation. Whether this satisfied this aim or not is up to the reader to decide. Secondly, this research aimed to place this notation into the relevant quantum cryptographic context. To this end, we worked out four protocols and their security diagrammatically, and also used the notation to represent a series of equivalences. In particular for quantum key recycling we developed new notation for eight state encoding, a concept that was recently introduced by Skoric and De Vries [4]. Furthermore we gave a security proof for eight state QKR in a similar fashion to how it was done for QKD in [6]. Finally, we worked out the findings of another publication [7] diagrammatically. In this research, the authors provide a series of security-wise equivalent steps, allowing them to prove security of one protocol indirectly by proving the security of another. We went through each of these steps diagrammatically, which allowed for both their formal proof and an intuitive understanding of these equivalences. Taking all of these findings into account, it can be said that this second aim - to build upon the novel notation and use it to add relevant research to the field of its role in quantum

cryptography - was successfully achieved. This work does leave open significant questions about the role of the diagrammatic method in the field of quantum cryptography, though. In future work that follows directly from this, a noisy version of QKR could be developed. In the larger context, the notation would benefit from being implemented in more protocols and their respective security proofs.

References

- [1] Abn amro investeert in quantumtechnologie, 2019.
- [2] Andor Buding. Race Against Time: Securing our Future Data with Quantum Encryption, 2015.
- [3] Bob Coecke and Aleks Kissinger. *Picturing Quantum Processes*. Cambridge University Press, Cambridge, 2017.
- [4] Manon de Vries and Boris Skoric. Quantum Key Recycling with Eight-State Encoding. *Cryptology ePrint Archive*, 2016.
- [5] Ivan Djordjevic. Chapter 7 - quantum error correction. In Ivan Djordjevic, editor, *Quantum Information Processing and Quantum Error Correction*, pages 227 – 276. Academic Press, Oxford, 2012.
- [6] Aleks Kissinger, Sean Tull, and Bas Westerbaan. Picture-perfect Quantum Key Distribution. 2017.
- [7] Daan Leermakers and Boris Škorić. Quantum Alice and Silent Bob. pages 1–13, 2019.
- [8] Daan Leermakers and Boris Skoric. Quantum alice and silent bob: Qubit-based quantum key recycling with almost no classical communication. Cryptology ePrint Archive, Report 2019/875, 2019. <https://eprint.iacr.org/2019/875>.
- [9] Göran Lindblad. A general no-cloning theorem. *Letters in Mathematical Physics*, 47(2):189–196, Jan 1999.
- [10] John Markoff. Codebook Shows an Encryption Form Dates Back to Telegraphs, 2011.
- [11] Michael a. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. 2011.
- [12] Mehdi Saeedi and Igor Markov. Synthesis and optimization of reversible circuits - a survey. *ACM Computing Surveys*, 45, 10 2011.
- [13] C. E. Shannon. Communication Theory of Secrecy Systems. *Bell System Technical Journal*, 1949.
- [14] Peter W. Shor and John Preskill. Simple proof of security of the BB84 quantum key distribution protocol. *Physical Review Letters*, 2000.
- [15] Stephen J. Weisner. Conjugate coding. *ACM SIGACT News*, 15(3):78–88, 1983.

A Keywords

Base: E, page 2
 Bra: A, page 1
 Colour: E, page 2
 Decoding: H, page 3
 Density matrix: G, page 3
 Discarding: G, page 3
 Doubling: F, page 2
 Effect: A, page 1
 Hermitian Adjoint: A, page 1
 Hermitian Conjugate: A, page 1
 Identity: B, page 1
 Ket: A, page 1
 Kronecker delta: D, page 2
 Linear map: C, page 1
 Phase: H, page 3
 Phase spider: H, page 3
 Purity: F and G, pages 2 and 3
 Spider: D, page 2
 State: A, page 1
 System type: B, page 1
 Tensor product: F, page 3
 Trace: G, page 3
 Map: C, page 1
 Mixed state: F, page 2
 Transpose: A, page 1
 Wire: B, page 1

B Eight state encryption map in Dirac notation

Here we will work out the eight state encryption map in Dirac notation. This will reveal also the reasoning behind its shape.

$$\text{⌞} : |g\rangle_z \mapsto |\psi_{uwg}\rangle \quad (157)$$

The operator that corresponds to this map, G_{uw} , is the following:

$$G_{uw} \equiv |0\rangle_z \langle \psi_{uw0}| + |1\rangle_z \langle \psi_{uw1}| \quad (158)$$

With its hermitian adjoint:

$$G_{uw}^\dagger \equiv |\psi_{uw0}\rangle \langle 0|_z + |\psi_{uw1}\rangle \langle 1|_z \quad (159)$$

Corresponding to the following map:

$$\text{⌞} : |g\rangle_z \mapsto |\psi_{uwg}\rangle \quad (160)$$

Since $G_{uw} \neq G_{uw}^\dagger$ it is not self-adjoint and due to the complex elements in $|\psi_{uwg}\rangle$ it is not self-conjugate either. We do however know that it is unitary:

$$G_{uw}G_{uw}^\dagger = |0\rangle_z \langle 0|_z + |1\rangle_z \langle 1|_z = \mathbb{I} \Leftrightarrow \text{---} \begin{array}{c} \text{---} \text{---} \\ \text{---} \text{---} \end{array} \text{---} = \text{---} \quad (161)$$