Lecture Notes

edX Quantum Cryptography: Week 0

Stephanie Wehner and Nelly Ng

Welcome to the lecture notes! Here you will find details as well as additional material for edX' "Quantum Cryptography". Week 0 is a - hopefully! - gentle introduction to quantum information. We will teach you all that you need to know to work with qubits and measurements mathematically. If you are curious how such qubits can be realized physically, or simply want more details than we provide here, we recommend [1] and [2].

0.1 Mathematical notation

Let us start by recalling commonly used definitions. For a complex number $c = a + ib \in \mathbb{C}$ with $a, b \in \mathbb{R}$ and $i = \sqrt{-1}$, we use $c^* = a - ib$ to denote its *complex conjugate*. We will also need mathematical notation that is used throughout quantum information. First, we will write vectors in a special way known as the "bra-ket" notation. While it may look a little cumbersome at first sight, it turns out to provide a convenient way of dealing with the many operations we will perform with such vectors. Let's start with two examples. We write $|v\rangle \in \mathbb{C}^2$ to denote a vector in a 2-dimensional vector space. For example,

$$|v\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} . \tag{1}$$

The "bra" of this vector is the conjugate transpose, which for our example looks like

$$\langle v| = \left((|0\rangle)^*\right)^T = \begin{pmatrix} 1^* \\ 0^* \end{pmatrix}^T = (1\ 0) . \tag{2}$$

The general definition of the "bra-ket" notation is as follows:

Definition 0.1.1 — Ket and Bra notation. A *ket*, denoted $|\cdot\rangle$, represents a d-dimensional column vector in the complex vector space $\mathbb{C}^d$. A *bra*, denoted $\langle\cdot|$, is a d-dimensional row vector equal to the complex conjugate of the corresponding ket, namely

$$\langle\cdot| = (|\cdot\rangle^*)^T, \tag{3}$$

where * denotes the entry-wise conjugate, and T denotes the transpose.

Since we work with complex numbers, we also introduce the *absolute value* of such numbers.

Definition 0.1.2 — Absolute value of a complex number. Consider a complex number $z \in \mathbb{C}$ expressed as $z = x + iy$ where $x, y \in \mathbb{R}$ are real numbers representing the real and imaginary parts of z respectively. The *absolute value*, or otherwise known as *modulus* of z is given by

$$|z| = \sqrt{z^* z} = \sqrt{x^2 + y^2}. \tag{4}$$

For example, for $z = 1 + i2$ its absolute value is given by $|z| = \sqrt{1^2 + 2^2} = \sqrt{5}$. Very frequently, we will need to compute the inner product of two vectors in the "bra-ket" notation. This inner product is defined as follows:

Definition 0.1.3 — Inner Product. Given two d-dimensional vectors

$$|v_1\rangle = \begin{pmatrix} a_1 \\ \vdots \\ a_d \end{pmatrix} \qquad \text{and} \qquad |v_2\rangle = \begin{pmatrix} b_1 \\ \vdots \\ b_d \end{pmatrix}, \tag{5}$$

their *inner product* is given by $\langle v_1|v_2\rangle := \langle v_1|\,|v_2\rangle = \sum_{i=1}^{d} a_i^* b_i$.

Note that the inner product of two vectors $|v_1\rangle, |v_2\rangle \in \mathbb{C}^d$ is in general a complex number. Later on, we shall see that the modulus squared of the inner product $|\langle v_1|v_2\rangle|^2$ is of much significance. As an example, let us consider the inner product of the vector $|v\rangle$ given in (2) and

$$|w\rangle = \begin{pmatrix} 2 \\ 3 \end{pmatrix} . \tag{6}$$

We have

$$\langle v|w\rangle = (1\ 0)\begin{pmatrix} 2 \\ 3 \end{pmatrix} = 1\cdot 2 + 0\cdot 3 = 2 . \tag{7}$$

Exercise 0.1.1 Show that $|\langle v_1|v_2\rangle|^2 = \langle v_1|v_2\rangle\langle v_2|v_1\rangle$. Hint: first, prove the relation $(\langle v_1|v_2\rangle)^* = \langle v_2|v_1\rangle$. ■

Quite frequently, we will care about the 2-norm, or more simply length, of a vector.

Definition 0.1.4 — Length of a ket vector. Consider a ket vector

$$|v\rangle = \begin{pmatrix} a_1 \\ \vdots \\ a_d \end{pmatrix} . \tag{8}$$

The length of $|v\rangle$ is given by

$$\| |v\rangle \|_2 = \sqrt{\langle v|v\rangle} = \sqrt{\sum_{i=1}^{d} a_i^* a_i} = \sqrt{\sum_{i=1}^{d} |a_i|^2} . \tag{9}$$

If $\| |v\rangle \|_2 = 1$ we say that $|v\rangle$ has norm 1, or simply, $|v\rangle$ is *normalized*.

■ **Example 0.1.1** Consider a ket $|v\rangle = \frac{1}{2}\begin{pmatrix} 1+i \\ 1-i \end{pmatrix} \in \mathbb{C}^2$. The corresponding bra is given by $\langle v| = \frac{1}{2}\begin{pmatrix} 1-i & 1+i \end{pmatrix}$, and the length of $|v\rangle$ is

$$\sqrt{\langle v|v\rangle} = \sqrt{\frac{1}{4}\cdot 2\cdot (1+i)(1-i)} = \sqrt{\frac{1}{2}(1+i-i-i^2)} = \sqrt{\frac{1}{2}\cdot 2} = 1. \tag{10}$$

■

We are assuming that your are familiar with the notion of an orthonormal basis from linear algebra. We will often write such a basis as $\{|b\rangle\}_b$. The condition of being orthonormal can be expressed succinctly as $\langle b|\hat{b}\rangle = \delta_{b\hat{b}}$ [1] for all $b, \hat{b}$.

0.2 What are qubits?

We are all intuitively familiar with the notion of bits in classical computing. How do quantum bits differ from classical bits? To see this let us start by writing classical bits somewhat differently. Instead of writing them as '0' and '1', let us first associate them with two vectors

$$0 \to |0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \text{ and } 1 \to |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} . \tag{11}$$

[1] $\delta_{ab} = 0$ if $a \neq b$, and $\delta_{ab} = 1$ for $a = b$.

Classical bits have many properties we take for granted, for example, we can copy them arbitrarily often. As we will see shortly, the same is not true in the quantum regime! Thinking of a physical implementation of bits, $|0\rangle$ and $|1\rangle$ could label the ground and excited state of an atom respectively. A bit can then be encoded by preparing the atom in the ground state (for $|0\rangle$) or the excited state (for $|1\rangle$). Many possible physical implementations of bits exist.

0.2.1 A single qubit

When thinking about vectors, it is indeed natural to ask whether we could have any vector $\alpha|0\rangle + \beta|1\rangle$. This is precisely the mathematical description of quantum bits. Instead of being just "0" and "1", quantum bits can be in a *superposition* between "0" and "1". Since "quantum bit" is somewhat long, researchers simply use the term "qubit" to refer to a quantum bit. Thinking of bits as vectors, a qubit can be described by a vector $|v\rangle \in \mathbb{C}^2$. The vector space $\mathbb{C}^2$ is also known as the *state space* of the qubit. An example of a qubit state is

$$|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \ . \tag{12}$$

Does any vector $|v\rangle \in \mathbb{C}^2$ form a valid qubit state? It turns out that in order to be a valid qubit, $|v\rangle$ must be normalized, just as the vectors $|0\rangle$ and $|1\rangle$ corresponding to classical bits were indeed normalized (check this for yourself!). For the moment, let us just take this as a rule, leading to the following definition.

Definition 0.2.1 — Qubit. A (pure) state of a *qubit* can be represented as a 2-dimensional ket vector $|\psi\rangle \in \mathbb{C}^2$,

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \ , \qquad \text{where } \alpha, \beta \in \mathbb{C} \text{ and } |\alpha|^2 + |\beta|^2 = 1. \tag{13}$$

The condition on α and β means that $|\psi\rangle$ is normalized. These complex numbers α and β are also called *amplitudes* of $|\psi\rangle$.

Throughout these lectures we will be mostly focusing on encoding information in qubits. However in general, quantum information can also be encoded in higher dimensional quantum systems. Therefore, one can similarly define a *qudit* as below:

Definition 0.2.2 — Qudit. A *qudit*, or a d-dimensional quantum system can be represented as a d-dimensional ket vector $|\psi\rangle \in \mathbb{C}^d$,

$$|\psi\rangle = \sum_{i=0}^{d-1} \alpha_i |i\rangle \ , \qquad \text{where } \forall i, \alpha_i \in \mathbb{C} \text{ and } \sum_{i=0}^{d-1} |\alpha_i|^2 = 1. \tag{14}$$

The condition on the coefficients α_i means that $|\psi\rangle$ is a vector of length of 1.

■ **Example 0.2.1** An example of a qubit is given by the vector $|-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$. The length of $|-\rangle$ is

$$\sqrt{\langle -|-\rangle} = \sqrt{\frac{1}{2}\begin{pmatrix}1 & -1\end{pmatrix}\begin{pmatrix}1\\-1\end{pmatrix}} = \sqrt{\frac{1}{2}\cdot 2} = 1, \tag{15}$$

so $|-\rangle$ is normalized. ■

Exercise 0.2.1 Verify that for all values of θ, $|\Psi\rangle = \cos(\theta)|0\rangle + \sin(\theta)|1\rangle$ is a valid qubit state. ■

In our definition of qubits, we started from a way to write classical bits as vectors $|0\rangle$ and $|1\rangle$. Note that these two vectors are orthonormal, which in the quantum notation can be expressed as $\langle 1|0\rangle = 0$ and $\langle 1|1\rangle = \langle 0|0\rangle = 1$. These two vectors thus form a basis for $\mathbb{C}^2$, in that any vector $|v\rangle \in \mathbb{C}^2$ can be written as $|v\rangle = \alpha|0\rangle + \beta|1\rangle$ for some coefficients $\alpha, \beta \in \mathbb{C}$. This basis corresponding to "classical" bits is used so often that it carries a special name:

Definition 0.2.3 — Standard basis. Consider the 2-dimensional complex vector space $\mathbb{C}^2$. The *standard basis*, or sometimes known as the *computational basis*, $\mathscr{S} = \{|0\rangle, |1\rangle\}$ is an orthonormal basis for this vector space, where the basis vectors are

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \text{ and } |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \tag{16}$$

Of course, there might be many other bases for $\mathbb{C}^2$. Another favorite basis which we will use rather frequently is the Hadamard basis defined as follows:

Definition 0.2.4 — Hadamard basis. The *Hadamard basis* is an orthonormal basis $\mathscr{H} = \{|+\rangle, |-\rangle\}$ consisting of the two basis elements

$$|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad \text{and } |-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ -1 \end{pmatrix}. \tag{17}$$

Let us verify that this is indeed an orthonormal basis using the "bra-ket" notation. As we have seen in Example 0.2.1, $|-\rangle$ is normalized. Similarly,

$$\langle +|+\rangle = \frac{1}{2}\begin{pmatrix} 1 & 1 \end{pmatrix}\begin{pmatrix} 1 \\ 1 \end{pmatrix} = \frac{1}{2}\cdot 2 = 1, \qquad \Longrightarrow, \sqrt{\langle +|+\rangle} = 1 \tag{18}$$

so $|+\rangle$ is also normalized. Furthermore, the inner product

$$\langle +|-\rangle = \frac{1}{2}\begin{pmatrix} 1 & 1 \end{pmatrix}\begin{pmatrix} 1 \\ -1 \end{pmatrix} = 0, \tag{19}$$

so $|+\rangle$ and $|-\rangle$ are orthogonal to each other.

Exercise 0.2.2 Express $|1\rangle$ in the Hadamard basis. That is, find coefficients α and β such that $|1\rangle = \alpha|+\rangle + \beta|-\rangle$. ■

0.2.2 Multiple qubits

Classically, if we have two bits, we write them as '00', '01' and so forth. But how can we write two qubits? One strategy is to again associate each of the two classical bits $x_1, x_2 \in \{0,1\}^2$ with a vector. Labelling the first qubit A and the second one B, we could perform the mapping from strings to orthonormal vectors as

$$0_A 0_B \to |00\rangle_{AB} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \qquad 0_A 1_B \to |01\rangle_{AB} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

$$1_A 0_B \to |10\rangle_{AB} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \qquad 1_A 1_B \to |11\rangle_{AB} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Note that the resulting vectors are in $\mathbb{C}^d$ with dimension $d = 2^2 = 4$, where the dimension corresponds to the number of possible strings. It turns out that one can write a two-qubit state $|\psi\rangle_{AB} \in \mathbb{C}^4$ as a superposition of these vectors, where we again demand that $|\psi_{AB}\rangle$ is normalized. As an example, let us consider a state $|\psi_{AB}\rangle$ that is an equal superposition of all the above standard basis vectors:

$$|\psi\rangle_{AB} = \frac{1}{2}|00\rangle_{AB} + \frac{1}{2}|01\rangle_{AB} + \frac{1}{2}|10\rangle_{AB} + \frac{1}{2}|11\rangle_{AB} \tag{20}$$

$$= \frac{1}{2}\begin{pmatrix}1\\0\\0\\0\end{pmatrix} + \frac{1}{2}\begin{pmatrix}0\\1\\0\\0\end{pmatrix} + \frac{1}{2}\begin{pmatrix}0\\0\\1\\0\end{pmatrix} + \frac{1}{2}\begin{pmatrix}0\\0\\0\\1\end{pmatrix} \tag{21}$$

$$= \frac{1}{2}\begin{pmatrix}1\\1\\1\\1\end{pmatrix}. \tag{22}$$

The sum of amplitudes $\frac{1}{2}$ squared is $4 \cdot \frac{1}{2^2} = 1$, therefore $|\psi\rangle$ is a valid two qubit quantum state. As you might have guessed, we now proceed analogously when considering n qubits. To address multiple qubits, we first look at the vector representation for multiple classical bits. For binary strings of length n, consider the vector space $\mathbb{C}^{2^n}$, where each coordinate is labelled by a string $x = x_1, \ldots, x_n$. There are a total of $d = 2^n$ such strings, so we can label each string x with a different integer $i \in [1, d]$. We can then express the string x as a vector $|x\rangle$ that is 0 everywhere, except at the position labelled by i. A quantum state of n qubits can then be written as

$$|\psi\rangle = \sum_{x \in \{0,1\}^n} \alpha_x |x\rangle \ , \tag{23}$$

with $\alpha_x \in \mathbb{C}$ and $\sum_x |\alpha_x|^2 = 1$. The numbers α_x are again called *amplitudes*. We emphasize that the dimension of the vector space $\mathbb{C}^{2^n}$ increases exponentially with the number n of bits. The space $\mathbb{C}^d$ with $d = 2^n$ is thereby called the *state space* of n qubits. This means that we need an exponential number of parameters α_x to keep track of only n qubits, in sharp contrast to the n parameters $x_1, \ldots, x_n$ to describe n classical bits.

You might wonder whether this was the only way to write down qubits. After all, we had simply chosen some mapping from strings of length n to vectors in $\mathbb{C}^d$. Could we have chosen any other mapping from strings to vectors? It turns out that the answer to this is yes - as long as each string gets mapped to a vector that is orthonormal to the others. The mapping above, however, is very convenient and generally adopted within the realm of quantum computing. Analogous to the case of a single qubit, the basis given by the set of vectors $\{|x\rangle \mid x \in \{0,1\}^n\}$ is called the *standard/computational basis*.

Definition 0.2.5 — Standard basis for n qubits. Consider the state space of n qubits $\mathbb{C}^d$, where $d = 2^n$. For each distinct string $x \in \{0,1\}^n$, associate x with a distinct integer $i \in \{1, 2, \cdots d\}$. The standard basis for $\mathbb{C}^d$ is an orthonormal basis given by $\mathscr{S}_n = \{|x\rangle\}_{x \in \{0,1\}^n}$,

where $|x\rangle$ are d-dimensional vectors

$$|x\rangle = \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix} \longrightarrow i\text{-th position.} \tag{24}$$

Let us summarize our discussion in the following definition of an n qubit quantum state.

Definition 0.2.6 An n-qubit state $|\psi\rangle \in \mathbb{C}^d$ with $d = 2^n$ can be written as a superposition of standard basis elements

$$|\psi\rangle = \sum_{x\in\{0,1\}^n} \alpha_x |x\rangle, \qquad \text{where } \forall x, \alpha_x \in \mathbb{C} \text{ and } \sum_{x\in\{0,1\}^n} |\alpha_x|^2 = 1. \tag{25}$$

Let us now consider two examples of two qubit states. The first is so famous it carries a special name and we will see it very frequently in the course of these notes.

■ **Example 0.2.2** Consider two qubits A and B, in the two qubit state known as the *EPR pair* [2], one can label the joint state as AB

$$|\text{EPR}\rangle_{AB} = \frac{1}{\sqrt{2}}(|00\rangle_{AB} + |11\rangle_{AB}) = \frac{1}{\sqrt{2}}\left(\begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}\right) = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}. \tag{26}$$

which is an equal superposition between the vectors $|00\rangle_{AB}$ and $|11\rangle$. The length of this vector is given by the (square root of) inner product

$$\langle \text{EPR}|\text{EPR}\rangle_{AB} = \frac{1}{\sqrt{2}}(\langle 00|_{AB} + \langle 11|_{AB}) \cdot \frac{1}{\sqrt{2}}(|00\rangle_{AB} + |11\rangle_{AB}) \tag{27}$$

$$= \frac{1}{2}(\underbrace{\langle 00|00\rangle_{AB}}_{1} + \underbrace{\langle 00|11\rangle_{AB}}_{0} + \underbrace{\langle 11|00\rangle_{AB}}_{0} + \underbrace{\langle 11|11\rangle_{AB}}_{1}) \tag{28}$$

$$= \frac{1}{2}\cdot 2 = 1, \qquad \Longrightarrow \qquad \sqrt{\langle EPR|EPR\rangle} = 1. \tag{29}$$

■

■ **Example 0.2.3** Consider the two qubit state

$$|\psi\rangle_{AB} = \frac{1}{\sqrt{2}}(|01\rangle_{AB} + |11\rangle_{AB}) = \frac{1}{\sqrt{2}}\begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix}. \tag{30}$$

For this state, the second qubit always corresponds to bit 1. We will later see that this is significantly different state compared to $|\text{EPR}\rangle_{AB}$ (hint: it is not entangled!). ■

0.3 Tensor Product: how to combine qubits

Let's imagine that we have two qubits, A and B. We know that we can describe the state of A as $|\psi\rangle_A$ and the one of B as $|\phi\rangle_B$. How can we write down the combined state $|\psi\rangle_{AB}$ of A *and* B

[2] The acronym EPR stands for Einstein, Podolsky and Rosen. Later we shall show that this state is entangled.

together? The rule for computing the joint state is given by the so-called tensor product (sometimes also called Kronecker product). For two qubits

$$|\psi\rangle_A = \alpha_A |0\rangle_A + \beta_A |1\rangle_A = \begin{pmatrix} \alpha_A \\ \beta_A \end{pmatrix}, \tag{31}$$

$$|\phi\rangle_B = \alpha_B |0\rangle_B + \beta_B |1\rangle_B = \begin{pmatrix} \alpha_B \\ \beta_B \end{pmatrix}, \tag{32}$$

the joint state $|\psi\rangle_{AB} \in \mathbb{C}^2 \otimes \mathbb{C}^2$ can be expressed as the tensor product of individual vectors $|\psi\rangle_A$ and $|\phi\rangle_B$

$$|\psi\rangle_{AB} = |\psi\rangle_A \otimes |\phi\rangle_B = \begin{pmatrix} \alpha_A \\ \beta_A \end{pmatrix} \otimes |\psi\rangle_B = \begin{pmatrix} \alpha_A |\psi\rangle_B \\ \beta_A |\psi\rangle_B \end{pmatrix} = \begin{pmatrix} \alpha_A \alpha_B \\ \alpha_A \beta_B \\ \beta_A \alpha_B \\ \beta_A \beta_B \end{pmatrix}. \tag{33}$$

As you may have guessed, we can of course also combine the state of two quantum systems A and B if they are larger than just one qubit. The general definition of the tensor product of two vectors is given by

Definition 0.3.1 Given two vectors $|\psi_1\rangle \in \mathbb{C}^{d_1}$ and $|\psi_2\rangle \in \mathbb{C}^{d_2}$ respectively, the tensor product is given by

$$|\psi_1\rangle \otimes |\psi_2\rangle = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_d \end{pmatrix} \otimes |\psi_2\rangle = \begin{pmatrix} \alpha_1 |\psi_2\rangle \\ \vdots \\ \alpha_d |\psi_2\rangle \end{pmatrix}, \tag{34}$$

and $|\psi_1\rangle \otimes |\psi_2\rangle$ lies in the state space $\mathbb{C}^{d_1} \otimes \mathbb{C}^{d_2}$.

The following simplified (or rather, lazy) notations are commonly used in quantum information:

$$\text{Omitting the tensor product symbol:} \quad |\psi\rangle_A \otimes |\psi\rangle_B = |\psi\rangle_A |\psi\rangle_B. \tag{35}$$

$$\text{Writing classical bits as a string:} \quad |0\rangle_A \otimes |0\rangle_B = |0\rangle_A |0\rangle_B = |00\rangle_{AB}. \tag{36}$$

$$\text{Combining several identical states:} \quad |\psi\rangle_1 \otimes |\psi\rangle_2 \cdots \otimes |\psi\rangle_n = |\psi\rangle^{\otimes n}. \tag{37}$$

Proposition 0.3.1 The tensor product satisfies several useful properties:

1. Distributive: $|\psi_1\rangle \otimes (|\psi_2\rangle + |\psi_3\rangle) = |\psi_1\rangle \otimes |\psi_2\rangle + |\psi_1\rangle \otimes |\psi_3\rangle$.
   Similarly, $(|\psi_1\rangle + |\psi_2\rangle) \otimes |\psi_3\rangle = |\psi_1\rangle \otimes |\psi_3\rangle + |\psi_2\rangle \otimes |\psi_3\rangle$.
2. Associative: $|\psi_1\rangle \otimes (|\psi_2\rangle \otimes |\psi_3\rangle) = (|\psi_1\rangle \otimes |\psi_2\rangle) \otimes |\psi_3\rangle$.
3. NOT commutative: In general, $|\psi_1\rangle \otimes |\psi_2\rangle \neq |\psi_2\rangle \otimes |\psi_1\rangle$ unless of course $|\psi_1\rangle = |\psi_2\rangle$.

These relations hold not only for kets, but also for bras.

To understand the definition of the tensor product, let us have a look at a few examples. The first relates to the definition of the standard basis for multiple qubits. Indeed, you may have been wondering, if we could have proceeded in a somewhat less ad hoc manner than starting from classical strings $x \in \{0,1\}^n$ and assigning to them vectors $|x\rangle$ in a space of dimension $d = 2^n$. Indeed, you may have started to wonder why n qubits resulted in a state space of a dimension that is exponential in n in the first place. The reason for this, is that the law of quantum mechanics tells us that the state space of two quantum systems is indeed combined by the tensor product.

■ **Example 0.3.1** Let's recover the standard basis of two qubits, from the standard basis of the individual qubits using the tensor product rule. Recall that the standard basis for two qubits AB is

given by

$$|00\rangle_{AB} = \begin{pmatrix}1\\0\\0\\0\end{pmatrix},\ |01\rangle_{AB} = \begin{pmatrix}0\\1\\0\\0\end{pmatrix},\ |10\rangle_{AB} = \begin{pmatrix}0\\0\\1\\0\end{pmatrix},\ |11\rangle_{AB} = \begin{pmatrix}0\\0\\0\\1\end{pmatrix}.$$

This basis can be constructed, by taking the tensor product of standard basis elements for individual qubits: $|0\rangle_A \otimes |0\rangle_B, |0\rangle_A \otimes |1\rangle_B, |1\rangle_A \otimes |0\rangle_B, |1\rangle_A \otimes |1\rangle_B$. For example, consider

$$|1\rangle_A \otimes |0\rangle_B = \begin{pmatrix}0\\1\end{pmatrix} \otimes |0\rangle_B = \begin{pmatrix}0\,|0\rangle_B\\1\,|0\rangle_B\end{pmatrix} = \begin{pmatrix}0\cdot 1\\0\cdot 0\\1\cdot 1\\1\cdot 0\end{pmatrix} = \begin{pmatrix}0\\0\\1\\0\end{pmatrix} = |10\rangle_{AB}. \tag{38}$$

■

We have already seen a few other examples of two qubit states. Let's see whether we can recover them from two individual qubit states using the tensor product.

■ **Example 0.3.2** Consider the states $|+\rangle_A = \frac{1}{\sqrt{2}}(|0\rangle_A + |1\rangle_A)$ and $|1\rangle_B$. The joint state $|\psi\rangle_{AB}$ is given by

$$|\psi\rangle_{AB} = |+\rangle_A \otimes |1\rangle_B = \frac{1}{\sqrt{2}}\begin{pmatrix}1\\1\end{pmatrix} \otimes |1\rangle_B = \frac{1}{\sqrt{2}}\begin{pmatrix}1\cdot|1\rangle_B\\1\cdot|1\rangle_B\end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix}0\\1\\0\\1\end{pmatrix}. \tag{39}$$

One can also express the joint state in the standard basis by:

$$\begin{aligned}|\psi\rangle_{AB} &= \frac{1}{\sqrt{2}}(|0\rangle_A + |1\rangle_A) \otimes |1\rangle_B && (40)\\ &= \frac{1}{\sqrt{2}}(|0\rangle_A \otimes |1\rangle_B + |1\rangle_A \otimes |1\rangle_B) && (41)\\ &= \frac{1}{\sqrt{2}}(|01\rangle_{AB} + |11\rangle_{AB}). && (42)\end{aligned}$$

This is the state we have seen in Example 0.2.3. ■

■ **Example 0.3.3** Consider the states $|+\rangle_A = \frac{1}{\sqrt{2}}(|0\rangle_A + |1\rangle_A)$ and $|+\rangle_B = \frac{1}{\sqrt{2}}(|0\rangle_B + |1\rangle_B)$. The joint state $|\psi\rangle_{AB}$ is

$$\begin{aligned}|\psi\rangle_{AB} &= \frac{1}{\sqrt{2}}(|0\rangle_A + |1\rangle_A) \otimes \frac{1}{\sqrt{2}}(|0\rangle_B + |1\rangle_B) && (43)\\ &= \frac{1}{2}(|00\rangle_{AB} + |01\rangle_{AB} + |10\rangle_{AB} + |11\rangle_{AB}) && (44)\\ &= \frac{1}{2}\begin{pmatrix}1\\1\\1\\1\end{pmatrix}. && (45)\end{aligned}$$

This is the state we have seen in (20), which is an equal superposition of all standard basis vectors for the two qubits. ■

The following is an example of a state that can actually not be expressed as the tensor product of two qubit states. Such states are rather special, and play an important role later in our course. Nevertheless, let's have a look at it to see how we might also express a two qubit state in different bases.

■ **Example 0.3.4** Consider the state

$$|\Psi\rangle_{AB} = \frac{1}{\sqrt{2}}(|+\rangle_A|+\rangle_B + |-\rangle_A|-\rangle_B). \tag{46}$$

Let us express this state in terms of the standard basis, by expanding the terms

$$|+\rangle_A|+\rangle_B = \frac{1}{2}(|0\rangle_A + |1\rangle_A)(|0\rangle_B + |1\rangle_B) = \frac{1}{2}(|00\rangle_{AB} + |10\rangle_{AB} + |01\rangle_{AB} + |11\rangle_{AB}) \tag{47}$$

$$|-\rangle_A|-\rangle_B = \frac{1}{2}(|0\rangle_A - |1\rangle_A)(|0\rangle_B - |1\rangle_B) = \frac{1}{2}(|00\rangle_{AB} - |10\rangle_{AB} - |01\rangle_{AB} + |11\rangle_{AB}). \tag{48}$$

Substituting this into Eq. (46) gives

$$|\Psi\rangle_{AB} = \frac{1}{\sqrt{2}}(|+\rangle_A|+\rangle_B + |-\rangle_A|-\rangle_B) \tag{49}$$

$$= \frac{1}{2\sqrt{2}}(|00\rangle_{AB} + |10\rangle_{AB} + |01\rangle_{AB} + |11\rangle_{AB} + |00\rangle_{AB} - |10\rangle_{AB} - |01\rangle_{AB} + |11\rangle_{AB}) \tag{50}$$

$$= \frac{1}{\sqrt{2}}(|00\rangle_{AB} + |11\rangle_{AB}) = |\text{EPR}\rangle_{AB} \tag{51}$$

where $|\text{EPR}\rangle_{AB}$ is the state we have seen previously in Example 0.2.2. We see that the coefficients of $|\text{EPR}\rangle_{AB}$ are the same whether we write it in the Hadamard basis or the standard basis. ■

0.4 Simple measurements

Let us consider what happens if we measure a qubit. Classically, you can think of the measurement of a bit as simply a readout: we have a system that encodes the state '0' and '1' and we make a measurement to find out which one it is.

0.4.1 Measurement in the standard basis

Let's first consider a single qubit. Quantum measurements can result in probabilistic outcomes, highlighting that quantum information and classical information really are fundamentally different. For example, if the state $|\psi\rangle \in \mathbb{C}^2$ is a superposition between $|0\rangle$ and $|1\rangle$, then upon measuring $|\psi\rangle$, we obtain different measurement outcomes corresponding to some probability distribution. How are such probabilities generated? The probability of different outcomes, for instance for outcome '0', can be computed by, roughly speaking, "looking at how much '0' is actually in our qubit vector". This is quantified by the inner product between $|\psi\rangle$ and $|0\rangle$. More concretely, consider a single qubit state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, where α, β are complex numbers. Upon measuring the qubit, one obtains the outcome "0" with probability p_0 and "1" with probability p_1. These probabilities can be determined by computing the inner products

$$p_0 = |\langle\psi|0\rangle|^2 = \left|\begin{pmatrix}\alpha^* & \beta^*\end{pmatrix}\begin{pmatrix}1\\0\end{pmatrix}\right|^2 = |\alpha|^2, \tag{52}$$

$$p_1 = |\langle\psi|1\rangle|^2 = \left|\begin{pmatrix}\alpha^* & \beta^*\end{pmatrix}\begin{pmatrix}0\\1\end{pmatrix}\right|^2 = |\beta|^2. \tag{53}$$

We now see a good reason for the condition $|\alpha|^2+|\beta|^2=1$: it means that $p_0+p_1=1$, that is, the probabilities of observing '0' and '1' add up to one. In quantum computer science, it is customary to label the outcomes '0' for "$|0\rangle$" and '1' for "$|1\rangle$" [3], while in physics people often use $+1$ for "$|0\rangle$" and -1 for "$|1\rangle$".

Application: Randomness from a deterministic process

Can we do anything interesting with what we have learned so far? It turns out the answer is yes: by preparing just single qubits and measuring in the standard basis, we can in principle achieve a task that it is impossible classically. Namely, we can produce true random numbers. Consider the following process illustrated in Figure 1: first, prepare a qubit in the state $|+\rangle=\frac{1}{\sqrt{2}}(|0\rangle+|1\rangle)$. Next, measure this state in the standard basis. The probability of obtaining each outcome can then

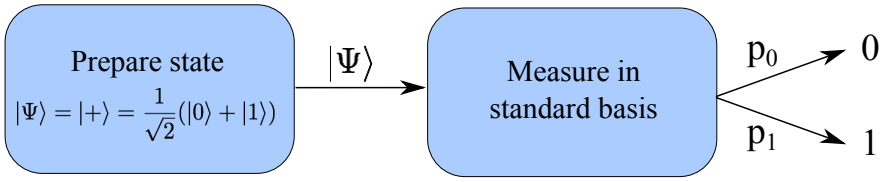

Figure 1: Generation of genuine randomness from the preparation of a qubit in superposition.

be calculated by evaluating the inner products:

$$p_0=|\langle+|0\rangle|^2=\left|\frac{1}{\sqrt{2}}(\langle 0|+\langle 1|)|0\rangle\right|^2=\left|\frac{1}{\sqrt{2}}(\underbrace{\langle 0|0\rangle}_{1}+\underbrace{\langle 1|0\rangle}_{0})\right|^2=\frac{1}{(\sqrt{2})^2}=\frac{1}{2}, \tag{54}$$

$$p_1=|\langle+|1\rangle|^2=\left|\frac{1}{\sqrt{2}}(\langle 0|+\langle 1|)|1\rangle\right|^2=\left|\frac{1}{\sqrt{2}}(\underbrace{\langle 0|1\rangle}_{0}+\underbrace{\langle 1|1\rangle}_{1})\right|^2=\frac{1}{(\sqrt{2})^2}=\frac{1}{2}. \tag{55}$$

This simple example already tells us something about the power of quantum information: We could build a machine that deterministically prepares the qubit $|+\rangle$, followed by a measurement in the standard basis. Since $p_0=p_1=1/2$, this machine allows us to produce a perfect random number - even though no randomness has been used inside our machine! In contrast, one can prove that no classical deterministic machine can produce random numbers from scratch.

We saw how to measure a single qubit in the standard basis. The rule for computing probabilities of measurement outcomes generalizes in a direct way to measuring n-qubit states. Indeed, consider an n-qubit quantum state

$$|\Psi\rangle=\sum_{x\in\{0,1\}^n}\alpha_x|x\rangle\ . \tag{56}$$

What happens when $|\Psi\rangle$ is measured in the standard basis $\{|x\rangle\}_x$? It turns out that the probability of outcome x is given by $p_x=|\langle x|\Psi\rangle|^2=|\alpha_x|^2$, explaining again the need for normalization of the vector $|\Psi\rangle$.

0.4.2 Measuring a qubit in other bases

Can we measure our qubit in any other basis? The answer to this is yes! Indeed this is another feature that distinguishes quantum from classical, where the only basis around is the standard basis. To find out how to analyze such a more general setting, let us first take a step back and consider how we found the probabilities above. When measuring in the standard basis, the probabilities are

[3] And more generally, x for outcomes "$|x\rangle$"

given by the squared amplitudes when writing out the state in terms of the standard basis. When measuring a qubit in a different orthonormal basis, given by vectors $\mathscr{G} = \{|v\rangle, |v^{\perp}\rangle\}$, it is intuitive that we would have to express the qubit in the new basis. That is, we need to find amplitudes $\hat{\alpha}$ and $\hat{\beta}$ such that

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = \hat{\alpha}|v\rangle + \hat{\beta}|v^{\perp}\rangle \ . \tag{57}$$

■ **Example 0.4.1** As an example, let consider again the qubit $|+\rangle = (1/\sqrt{2})(|0\rangle + |1\rangle)$. Instead of measuring it in the standard basis, let us now measure in the basis $\mathscr{H} = \{|+\rangle, |-\rangle\}$ given by the two orthonormal vectors $|+\rangle$ and $|-\rangle = (1/\sqrt{2})(|0\rangle - |1\rangle)$. Clearly, we can write the qubit as $1 \cdot |+\rangle + 0 \cdot |-\rangle$. Thus the probability of obtaining measurement outcome "$|+\rangle$" is 1. We thus see that the probabilities of measurement outcomes depends dramatically on the basis in which we measure. ■

■ **Example 0.4.2** Consider measuring an arbitrary qubit $\alpha|0\rangle + \beta|1\rangle$ in the basis $\{|+\rangle, |-\rangle\}$. To find out how to express the qubit in this other basis, it is convenient to determine how the basis elements $|0\rangle$ and $|1\rangle$ look like in this basis. We find that

$$|0\rangle = \frac{1}{2}[(|0\rangle + |1\rangle) + (|0\rangle - |1\rangle)] = \frac{1}{\sqrt{2}}(|+\rangle + |-\rangle) \ , \tag{58}$$

$$|1\rangle = \frac{1}{2}[(|0\rangle + |1\rangle) - (|0\rangle - |1\rangle)] = \frac{1}{\sqrt{2}}(|+\rangle - |-\rangle) \ . \tag{59}$$

We thus have

$$\alpha|0\rangle + \beta|1\rangle = \frac{1}{\sqrt{2}}[\alpha(|+\rangle + |-\rangle) + \beta(|+\rangle - |-\rangle)] = \tag{60}$$

$$= \frac{\alpha+\beta}{\sqrt{2}}|+\rangle + \frac{\alpha-\beta}{\sqrt{2}}|-\rangle \ . \tag{61}$$

This means that we obtain outcome "$|+\rangle$" with probability $|\alpha+\beta|^2/2$ and outcome "$|-\rangle$" with probability $|\alpha-\beta|^2/2$. ■

Exercise 0.4.1 Consider the state $|\Psi\rangle = |0\rangle$. What are the probabilities p_0, p_1 for measuring $|\Psi\rangle$ in the standard basis? What are the probabilities p_+, p_- for measuring $|\Psi\rangle$ in the Hadamard basis? ■

Quite often we do not care about the entire probability distribution, but just the probability of one specific outcome. Is there a more efficient way to find this probability than to rewrite the entire state $|\psi\rangle$ in another basis? To investigate this, let us consider a single qubit

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \ . \tag{62}$$

Remember that the elements of the standard basis are orthonormal. This means that

$$(|0\rangle)^{\dagger}|0\rangle = \begin{pmatrix} 1 & 0 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \end{pmatrix} = 1 \ , \tag{63}$$

$$(|0\rangle)^{\dagger}|1\rangle = \begin{pmatrix} 1 & 0 \end{pmatrix}\begin{pmatrix} 0 \\ 1 \end{pmatrix} = 0 \ . \tag{64}$$

Because the vectors are orthonormal, we could thus have found the desired probabilities by simply computing the inner product between two vectors, as claimed above. Specifically, when given the

qubit $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ we obtain outcomes "$|0\rangle$" and "$|1\rangle$" with probabilities

$$p_0 = |\langle 0|\psi\rangle|^2 = \left|(1\ 0)\begin{pmatrix}\alpha\\ \beta\end{pmatrix}\right|^2 = |\alpha|^2 \tag{65}$$

$$p_1 = |\langle 1|\psi\rangle|^2 = \left|(0\ 1)\begin{pmatrix}\alpha\\ \beta\end{pmatrix}\right|^2 = |\beta|^2 \tag{66}$$

$$\tag{67}$$

■ **Example 0.4.3** Suppose we measure $|0\rangle$ in the Hadamard basis $\mathscr{H}$ (see above). The probabilities of observing outcomes "$|+\rangle$" and "$|-\rangle$" are given by

$$p_+ = |\langle +|0\rangle|^2 = \left|(1/\sqrt{2}\ 1/\sqrt{2})\begin{pmatrix}1\\ 0\end{pmatrix}\right|^2 = \frac{1}{2}\,, \tag{68}$$

$$p_- = |\langle -|0\rangle|^2 = \left|(1/\sqrt{2}\ -1/\sqrt{2})\begin{pmatrix}1\\ 0\end{pmatrix}\right|^2 = \frac{1}{2}\,. \tag{69}$$

■

For multiple qubits, the rule for finding probabilities is analogous.

Definition 0.4.1 Suppose that we measure a quantum state $|\psi\rangle$ in the orthonormal basis $\{|b_j\rangle\}_{j=1}^d$. The probability of observing outcome "b_j" can be found by computing

$$p_j = |\langle b_j|\psi\rangle|^2\,. \tag{70}$$

The post-measurement state when obtaining outcome "b_j" is given by $|b_j\rangle$.

Let us now consider some examples to gain intuition on measuring quantum systems in different bases. First, let us have a look at a single qubit example.

■ **Example 0.4.4** Consider the qubit $|\Psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle)$, and measure the qubit in the $\{|+\rangle, |-\rangle\}$ basis. The probabilities of obtaining "+" and "–" can be evaluated as follows:

$$p_+ = |\langle\Psi|+\rangle|^2 = \left|\frac{1}{2}(\langle 0| - i\langle 1|)(|0\rangle + |1\rangle)\right|^2 \tag{71}$$

$$= \frac{1}{4}\left|\langle 0|0\rangle + \langle 0|1\rangle - i\langle 1|0\rangle - i\langle 1|1\rangle\right|^2 \tag{72}$$

$$= \frac{1}{4}|1-i|^2 \tag{73}$$

$$= \frac{1}{4}(1-i)(1+i) = \frac{1}{2}, \tag{74}$$

$$p_- = |\langle\Psi|-\rangle|^2 = \left|\frac{1}{2}(\langle 0| - i\langle 1|)(|0\rangle - |1\rangle)\right|^2 \tag{75}$$

$$= \frac{1}{4}\left|\langle 0|0\rangle - \langle 0|1\rangle - i\langle 1|0\rangle + i\langle 1|1\rangle\right|^2 \tag{76}$$

$$= \frac{1}{4}|1+i|^2 \tag{77}$$

$$= \frac{1}{4}(1+i)(1-i) = \frac{1}{2}, \tag{78}$$

This example shows that when the states involved have complex-valued amplitudes, one has to take extra caution when evaluating the inner product: namely when taking the bra $\langle\Psi|$, one should

remember to alter the +/- sign whenever a complex number is involved (since the bra $\langle\Psi|$ is the conjugate transpose of the ket $|\Psi\rangle$). ■

While we will generally talk about n-qubits, we can of course also consider a quantum system comprised of three levels $|0\rangle$, $|1\rangle$, and $|2\rangle$, i.e. a qutrit. The rule for obtaining the probabilities of measurement outcomes remains unchanged.

■ **Example 0.4.5** Consider a *qutrit*, which is a 3-dimensional quantum system represented by the vector

$$|v\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}1\\0\\0\end{pmatrix} + \frac{1}{2}\begin{pmatrix}0\\1\\0\end{pmatrix} + \frac{1}{2}\begin{pmatrix}0\\0\\1\end{pmatrix}, \tag{79}$$

and measure in the basis $\mathscr{B} = \{|b_1\rangle, |b_2\rangle, |b_3\rangle\}$ where

$$|b_1\rangle = \begin{pmatrix}1\\0\\0\end{pmatrix}, \qquad |b_2\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}0\\1\\1\end{pmatrix}, \qquad |b_3\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}0\\1\\-1\end{pmatrix}. \tag{80}$$

The probabilities of obtaining each outcome can be calculated as follows:

$$p_{b_1} = |\langle b_1|v\rangle|^2 = \frac{1}{2}, \tag{81}$$

$$p_{b_2} = |\langle b_2|v\rangle|^2 = \langle b_2|v\rangle\langle v|b_2\rangle = \frac{1}{2\sqrt{2}}(1+1)\cdot\frac{1}{2\sqrt{2}}(1+1) = \frac{1}{2}, \tag{82}$$

$$p_{b_3} = |\langle b_3|v\rangle|^2 = \langle b_3|v\rangle\langle v|b_3\rangle = \frac{1}{2\sqrt{2}}(1-1)\cdot\frac{1}{2\sqrt{2}}(1-1) = 0. \tag{83}$$

■

Expectation values

Physicists (but also computer scientists!) like to compute expectation values of measurement outcomes, as they provide an indication of the average behavior, if one was to perform a measurement many times (however we shall see later, that the measurement will perturb the state!). Let us suppose that we measure a qubit $|\Psi\rangle$ in the standard basis $\{|0\rangle, |1\rangle\}$. We will also adopt the physics convention of labelling these outcomes ± 1. This means that we associate the outcome "$|0\rangle$" with outcome $+1$, and outcome "$|1\rangle$" with outcome -1. The expectation value the outcome obtained when measuring $|\Psi\rangle$ is then

$$E = 1\cdot|\langle 0|\psi\rangle|^2 - 1\cdot|\langle 1|\psi\rangle|^2. \tag{84}$$

Note that since $|\langle 0|\psi\rangle|^2 = \langle\psi|0\rangle\langle 0|\psi\rangle$, we have

$$E = \langle\psi|\left(|0\rangle\langle 0| - |1\rangle\langle 1|\right)|\psi\rangle = \langle\psi|Z|\psi\rangle \tag{85}$$

where $Z = |0\rangle\langle 0| - |1\rangle\langle 1|$. As we shall see later, Z is called the Pauli-Z matrix.

0.4.3 Measuring multiple systems

We saw how to measure some quantum state $|\psi\rangle$. Let us now consider what happens if we measure the state of multiple qubits, where we think of measuring each qubit in a separate basis. To understand this, it is useful to realize that a basis for the joint state space $\mathbb{C}_A^{d_A}\otimes\mathbb{C}_B^{d_B}$ can be obtained from bases for the individual state spaces $\mathbb{C}_A^{d_A}$ and $\mathbb{C}_B^{d_B}$. Specifically, if $\{|b_j^A\rangle\}_j$ is a basis for $\mathbb{C}_A^{d_A}$ and $\{|b_j^B\rangle\}_j$ is a basis for the state space $\mathbb{C}_B^{d_B}$, then the set of vectors $\{\{|b_j^A\rangle\otimes|b_k^B\rangle\}_{j=1}^{d_A}\}_{k=1}^{d_B}$ gives a basis for $\mathbb{C}_A^{d_A}\otimes\mathbb{C}_B^{d_B}$.

■ **Example 0.4.6** Consider the basis $\{|0\rangle_A, |1\rangle_A\}$ for qubit A, and the basis $\{|+\rangle_B, |-\rangle_B\}$ for qubit B. A basis for the joint state AB is then given by $\{|0\rangle_A|+\rangle_B, |0\rangle_A|-\rangle_B, |1\rangle_A|+\rangle_B, |1\rangle_A|-\rangle_B\}$. ■

Let us now think how we might construct some measurement for two quantum states from measurements of the individual ones. Suppose we measure particle A in the basis $\{|b_j^A\rangle\}_j$ and particle B in the basis $\{|b_k^B\rangle\}_k$ when the joint state of both particles is given by $|\psi\rangle_{AB}$. What is the probability that we obtain outcome " $|b_j^A\rangle$ " on A, and outcome " $|b_k^B\rangle$ " on B? To find such joint probabilities, we first write down the joint basis of quantum states A and B as above: $\{\{|b_j^A\rangle|b_k^B\rangle\}_j\}_k$. We can then apply the usual rule to compute the probability as

$$p_{jk} = |\langle b_j^A|\langle b_k^B||\psi\rangle_{AB}|^2 \,. \tag{86}$$

■ **Example 0.4.7** Consider two qubits in an EPR pair

$$|\text{EPR}\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle), \tag{87}$$

and measure them both in the standard basis. The probabilities of obtaining outcomes 00, 01, 10, and 11 are given by

$$p_{00} = p_{11} = \frac{1}{2}, \tag{88}$$

$$p_{01} = p_{10} = 0. \tag{89}$$

■

0.5 Transformations on qubits

Just like on classical bits, we can perform operations on qubits. Since we can write quantum states as vectors, we are looking for a linear operator U that maps vectors to vectors

$$|\psi_{\text{out}}\rangle = U|\psi_{\text{in}}\rangle \tag{90}$$

for some matrix U. If $|\psi_{\text{in}}\rangle \in \mathbb{C}^d$, then U is a $d \times d$ matrix with complex entries. Recall that for any quantum state we have $\langle\psi|\psi\rangle = 1$. And we have also seen that this is quite important, because it tells us that the sum of the probabilities, if we measure the state, should also be 1. This means that the operation U should preserve the inner product [4], i.e.,

$$\langle\psi_{\text{out}}|\psi_{\text{out}}\rangle = \langle\psi_{\text{in}}|U^\dagger U|\psi_{\text{in}}\rangle = 1 \,. \tag{91}$$

Similarly, the same should be true for the operation $U^\dagger$

$$\langle\psi_{\text{out}}|\psi_{\text{out}}\rangle = \langle\psi_{\text{in}}|UU^\dagger|\psi_{\text{in}}\rangle = 1 \,. \tag{92}$$

We see that in order to preserve probabilities the operation U should preserve the length of any vector. This is the case precisely if $U^\dagger U = UU^\dagger = \mathbb{I}$, where $\mathbb{I}$ is the identity matrix. Such a matrix $\mathbb{I}$ will continually appear throughout these notes, and we define it below.

Definition 0.5.1 — Identity. The identity $\mathbb{I}$ is a diagonal, square matrix where each diagonal element is equal to 1, i.e.

$$\mathbb{I} = \begin{pmatrix} 1 & 0 & \cdots & \cdots & 0 \\ 0 & 1 & \cdots & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 1 \end{pmatrix}. \tag{93}$$

[4] Remember that $(U|\psi\rangle)^\dagger = \langle\psi|U^\dagger$.

For any dimension d, we denote the $d \times d$ identity matrix as $\mathbb{I}_d$.

R The identity matrix is a unitary operation that preserves all quantum states, i.e. for any quantum state $|\psi\rangle$, $\mathbb{I}|\psi\rangle = |\psi\rangle$.

We will typically not specify the dimension of the identity matrix explicitly if it can be inferred from context. The only allowed operations in the quantum regime are unitary operations.

Definition 0.5.2 — Unitary operation. An operation U is unitary if and only if $U^\dagger U = UU^\dagger = \mathbb{I}$.

To gain some intuition about unitary operations, let us have a look at some useful examples.

■ **Example 0.5.1** Consider the matrix

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}. \tag{94}$$

You can convince yourself that $H^\dagger = H$ and thus

$$H^\dagger H = HH = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \mathbb{I}\,. \tag{95}$$

That is, H is unitary. We have that

$$H|0\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = |+\rangle\ . \tag{96}$$

Similarly, you can convince yourself that $H|1\rangle = |-\rangle$. We thus see that H transforms the computational basis $\{|0\rangle, |1\rangle\}$ into the Hadamard basis $\{|+\rangle, |-\rangle\}$. Indeed, H is called the *Hadamard transform*. ■

Note that $\mathbb{I}$ is itself also a unitary operation, called the *identity operation*. It just means that the state is not transformed at all. Let us now consider a somewhat more complicated operation.

■ **Example 0.5.2** For any $\theta \in \mathbb{R}$, consider the matrix

$$R(\theta) = \begin{pmatrix} \cos\frac{\theta}{2} & -\sin\frac{\theta}{2} \\ \sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{pmatrix}. \tag{97}$$

The adjoint of this matrix is given by

$$R^\dagger(\theta) = \begin{pmatrix} \cos\frac{\theta}{2} & \sin\frac{\theta}{2} \\ -\sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{pmatrix}, \tag{98}$$

and therefore

$$R(\theta)R^\dagger(\theta) = \begin{pmatrix} \cos\frac{\theta}{2} & -\sin\frac{\theta}{2} \\ \sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{pmatrix} \cdot \begin{pmatrix} \cos\frac{\theta}{2} & \sin\frac{\theta}{2} \\ -\sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{pmatrix} \tag{99}$$

$$= \begin{pmatrix} \cos^2\frac{\theta}{2} + \sin^2\frac{\theta}{2} & 0 \\ 0 & \sin^2\frac{\theta}{2} + \cos^2\frac{\theta}{2} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \tag{100}$$

One can check that $R^\dagger(\theta)R(\theta) = \mathbb{I}$ as well, therefore $R(\theta)$ is unitary.

$$R(\theta)|0\rangle = \begin{pmatrix} \cos\frac{\theta}{2} & -\sin\frac{\theta}{2} \\ \sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} \cos\frac{\theta}{2} \\ \sin\frac{\theta}{2} \end{pmatrix}. \tag{101}$$

$$R(\theta)|1\rangle = \begin{pmatrix} \cos\frac{\theta}{2} & -\sin\frac{\theta}{2} \\ \sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} -\sin\frac{\theta}{2} \\ \cos\frac{\theta}{2} \end{pmatrix}. \tag{102}$$

If we take $\theta = \frac{\pi}{2}$, then $\cos\frac{\theta}{2} = \sin\frac{\theta}{2} = \cos\frac{\pi}{4} = \frac{1}{\sqrt{2}}$ and therefore

$$R\left(\frac{\pi}{2}\right)|0\rangle = |+\rangle \qquad \text{and} \qquad R\left(\frac{\pi}{2}\right)|1\rangle = -|-\rangle\,. \tag{103}$$

■

0.5.1 Pauli matrices as unitary operations

In this section we look at the Pauli matrices, commonly denoted as X, Y, Z. These are quite famous in physics, but also have rather interesting interpretations as bit and phase flip operations as we will see below. The Pauli matrices are unitary 2×2 matrices, with the following form

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \tag{104}$$

$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \tag{105}$$

$$Y = iXZ. \tag{106}$$

The Pauli-X matrix acts on the standard basis vectors by interchanging them:

$$X|0\rangle = |1\rangle\ , \tag{107}$$

$$X|1\rangle = |0\rangle\ . \tag{108}$$

In analogy to classical computation X is also referred to as NOT, since it changes 0 to 1 and vice versa. This is also known as a *bit flip* operation. On the other hand, the Pauli-Z matrix acts on the standard basis by introducing a *phase flip*

$$Z|0\rangle = |0\rangle\ , \tag{109}$$

$$Z|1\rangle = -|1\rangle\ . \tag{110}$$

The Pauli-Z matrix has the effect of interchanging the vectors $|+\rangle$ and $|-\rangle$. To be precise, we have

$$Z|+\rangle = Z(|0\rangle + |1\rangle)/\sqrt{2} = (Z|0\rangle + Z|1\rangle)/\sqrt{2} = (|0\rangle - |1\rangle)/\sqrt{2} = |-\rangle\,. \tag{111}$$

Similarly, $Z|-\rangle = |+\rangle$. We thus see that Z acts like a bit flip upon the Hadamard basis, while it acts like a phase flip in the standard basis. Applying both a bit and a phase flip gives $Y = iXZ$. The i makes Y Hermitian, that is, $Y^\dagger = Y$. This matrix, when acted upon the standard basis vectors, introduces a bit flip and a phase flip:

$$Y|0\rangle = iXZ|0\rangle = iX|0\rangle = i|1\rangle\,. \tag{112}$$

$$Y|1\rangle = -iXZ|0\rangle = -iX|1\rangle = -i|0\rangle\,. \tag{113}$$

Exercise 0.5.1 Verify that the Pauli matrices X, Z and Y are indeed unitary. ■

0.6 No cloning!

In this section we show that arbitrary qubits, unlike classical bits, cannot be copied. Here, we provide a slightly different proof than shown in the lecture. If we did have a copying unitary C it

should give us $C(|\psi\rangle \otimes |0\rangle) = |\psi\rangle \otimes |\psi\rangle$ for *any* input qubit $|\psi\rangle$. By contradiction, let us suppose such a unitary existed. In particular, such a unitary gives us

$$C(|\psi_1\rangle \otimes |0\rangle) = |\psi_1\rangle \otimes |\psi_1\rangle \tag{114}$$
$$C(|\psi_2\rangle \otimes |0\rangle) = |\psi_2\rangle \otimes |\psi_2\rangle \tag{115}$$

Since C is a unitary, we have $C^\dagger C = \mathbb{I}$ and hence

$$\langle\psi_1|\psi_2\rangle = \langle\psi_1|\psi_2\rangle\langle 0|0\rangle \tag{116}$$
$$= (\langle\psi_1| \otimes \langle 0|)C^\dagger C(|\psi_2\rangle \otimes |0\rangle) \tag{117}$$
$$= (\langle\psi_1| \otimes \langle\psi_1|)(|\psi_2\rangle \otimes |\psi_2\rangle) = (\langle\psi_1|\psi_2\rangle)^2. \tag{118}$$

Clearly whenever $0 < |\langle\psi_1|\psi_2\rangle| < 1$, the above cannot hold and hence such a copying unitary C cannot exist. Note that $|\psi_1\rangle = |0\rangle$ and $|\psi_2\rangle = |+\rangle$, for example, have precisely this property.

The fact that we cannot clone, that is, copy arbitrary quantum states shows that quantum information really is very different from classical information. It also allows us to gain further understanding: note that this also means that we cannot determine α and β from a single copy of a qubit $|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle$. Otherwise, we could copy the qubit by making a machine that prepares a qubit in the superposition $\alpha|0\rangle + \beta|1\rangle$.

While this has some nice features - for example, an inbuilt copy protection mechanism - it also means that qubits are very precious. When trying to send a qubit for example, we cannot simply try again when we failed such as with classical bits. If you could not hear me in the videos clearly, you could rewind, turn up the volume and try again. If I were talking qubits to you, there would be no way to do that!

0.7 Bloch sphere

For single qubits, there is a very convenient visual representation in terms of the so-called Bloch sphere. It should be noted that such a nice representation only exists for single qubits. To make this work, express the qubit as

$$|\psi\rangle = e^{i\gamma}\left(\cos\frac{\theta}{2}|0\rangle + e^{i\phi}\sin\frac{\theta}{2}|1\rangle\right), \tag{119}$$

where γ, θ and ϕ are real numbers. The global phase $e^{i\gamma}$ is neglected, since it has no observable effects on the probability of measurement outcomes. To see this, consider the states

$$|\psi_1\rangle = e^{i\gamma_1}\left(\cos\frac{\theta}{2}|0\rangle + e^{i\phi}\sin\frac{\theta}{2}|1\rangle\right), \tag{120}$$
$$|\psi_2\rangle = e^{i\gamma_2}\left(\cos\frac{\theta}{2}|0\rangle + e^{i\phi}\sin\frac{\theta}{2}|1\rangle\right), \tag{121}$$

for some real numbers γ_1, γ_2. Note that $|\psi_1\rangle = e^{i(\gamma_1-\gamma_2)}|\psi_2\rangle$. We thus have that for any measurement with respect to a basis $\{|b\rangle\}_b$, the probability of obtaining any outcome b is equal for both states, since

$$|\langle\psi_1|b\rangle|^2 = \langle b|\psi_1\rangle\langle\psi_1|b\rangle = e^{i(\gamma_1-\gamma_2)}e^{-i(\gamma_1-\gamma_2)}\langle b|\psi_2\rangle\langle\psi_2|b\rangle = |\langle\psi_2|b\rangle|^2. \tag{122}$$

Also, note that this parametrization preserves the normalization condition since $|\alpha|^2 + |\beta|^2 = \cos^2(\theta/2) + \sin^2(\theta/2) = 1$. In terms of the numbers (θ, ϕ) we can thus think of the qubit as a point on a 3 dimensional sphere as in Figure 2. It should be emphasized that this sphere does not follow the same coordinates as we have used for the vectors $|v\rangle \in \mathbb{C}^2$, but rather we need to translate to this new coordinate system.

Definition 0.7.1 The parametrization (θ, ϕ) of

$$|\psi\rangle = e^{i\gamma}\left(\cos\frac{\theta}{2}|0\rangle + e^{i\phi}\sin\frac{\theta}{2}|1\rangle\right) \tag{123}$$

is called the *Bloch sphere representation* (Figure 2) and a qubit can be described by a *Bloch vector* $\vec{r} = (\cos\phi\sin\theta, \sin\phi\sin\theta, \cos\theta)$.

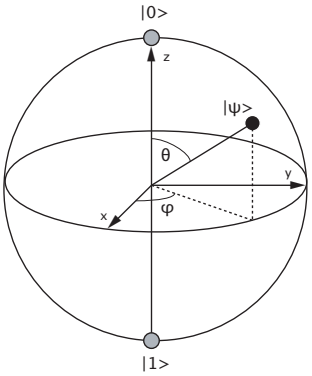

Figure 2: Bloch Sphere

Consider a qubit in the representation of Eq. (119) where $\gamma = \phi = 0$. Then the Bloch sphere representation of such a qubit lies on the xz-plane. The usefulness of this representation becomes immediately apparent when we consider the effects of the Hadamard transform on a qubit. Note that $(|0\rangle + |1\rangle)/\sqrt{2}$ can be found in Figure 2 at the intersection of the positive x-axis and the sphere. It is then easy to see that we can describe the effect of H on $(|0\rangle + |1\rangle)/\sqrt{2}$ as a rotation around the y-axis towards $|1\rangle$, followed by a reflection in the xy-plane. In fact, the Bloch sphere representation allows one to view all single qubit operations as rotations on this sphere. While we will make little use of this in this class, it is interesting to see how single qubit unitaries U can be expressed as rotations on the Bloch sphere. A rotation matrix $R_s(\theta)$ is a unitary operation that rotates a qubit Bloch vector around the axes $s \in \{x, y, z\}$ by an angle θ. Such matrices have the following form:

$$R_x(\theta) = e^{-i\theta X/2},\ R_y(\theta) = e^{-i\theta Y/2} \text{ and } R_z(\theta) = e^{-i\theta Z/2}, \tag{124}$$

where X, Y, Z are the Pauli matrices. Especially important for this text will be the rotation around the z axis. We can express it in more detail as

$$R_z(\theta) = e^{-i\theta Z/2} = \begin{pmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{pmatrix} = e^{-i\theta/2}\begin{pmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{pmatrix}.$$

Any arbitrary single qubit operation U can be expressed in terms of these rotations as

$$U = e^{i\alpha}R_z(\beta)R_y(\gamma)R_z(\delta)$$

for some real numbers α, β, γ and δ.

Lastly, it is worth noting that such a clean and simple representation only holds for the case of single qubits: for higher dimensional systems, it is not possible to represent a qudit in terms of a d-dimensional sphere!

Important identities for calculations

Given two vectors $|v_1\rangle = \begin{pmatrix} a_1 & \cdots & a_d \end{pmatrix}^T$ and $|v_2\rangle = \begin{pmatrix} b_1 & \cdots & b_d \end{pmatrix}^T$,

1. **(Inner product)** $\langle v_1|v_2\rangle := \langle v_1|\,|v_2\rangle = \sum_{i=1}^{d} a_i^* b_i$.
2. **(Tensor Product)**

$$|v_1\rangle \otimes |v_2\rangle := \begin{pmatrix} a_1b_1 & a_1b_2 & \cdots & a_1b_d & a_2b_1 & \cdots & a_1b_d & \cdots & a_db_d \end{pmatrix}^T.$$

Commonly used orthonormal bases for qubits

Standard basis for 1 qubit: $\mathscr{S} = \{|0\rangle, |1\rangle\}$ where $|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ and $|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$.
Standard basis for n qubits: $\mathscr{S}_n = \{|x\rangle\}_{x\in\{0,1\}^n}$ where for any string $x = x_1x_2\cdots x_n$, $|x\rangle = |x_1\rangle \otimes |x_2\rangle \otimes \cdots \otimes |x_n\rangle$.
Hadamard basis for 1 qubit: $\mathscr{H} = \{|+\rangle, |-\rangle\}$ where $|\pm\rangle = \frac{1}{\sqrt{2}}(|0\rangle \pm |1\rangle)$. Since these are orthonormal bases, the following holds:

$$\langle 0|1\rangle = \langle 1|0\rangle = 0, \qquad \langle 0|0\rangle = \langle 1|1\rangle = 1, \tag{125}$$
$$\langle +|-\rangle = \langle -|+\rangle = 0, \qquad \langle +|+\rangle = \langle -|-\rangle = 1, \tag{126}$$
$$\langle x|x'\rangle = \delta_{xx'}, \text{ where } x, x' \in \{0,1\}^n \text{ and } \delta_{xx'} \text{ is the Kronecker-delta function.} \tag{127}$$

Common representations of a qubit

Standard representation: $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, where $\alpha, \beta \in \mathbb{C}$.
Bloch sphere representation: $|\psi\rangle = e^{i\gamma}\left(\cos\frac{\theta}{2}|0\rangle + e^{i\phi}\sin\frac{\theta}{2}|1\rangle\right)$, where $\gamma, \theta, \phi \in \mathbb{R}$.

Properties of the tensor product

For any $|v_1\rangle, |v_2\rangle$ and $|v_3\rangle$,

1. Distributive: $|v_1\rangle \otimes (|v_2\rangle + |v_3\rangle) = |v_1\rangle \otimes |v_2\rangle + |v_1\rangle \otimes |v_3\rangle$
   Also, $|v_1\rangle \otimes (|v_2\rangle + |v_3\rangle) = |v_1\rangle \otimes |v_3\rangle + |v_2\rangle \otimes |v_3\rangle$.
2. Associative: $(|v_1\rangle + |v_2\rangle) \otimes |v_3\rangle) = (|v_1\rangle \otimes |v_2\rangle) \otimes |v_3\rangle$.

Similarly, these relations hold for any $\langle v_1|, \langle v_2|$ and $\langle v_3|$.

Probability of measurement outcomes

Consider measuring a quantum state $|\Psi\rangle$ in an orthonormal basis $\mathscr{B} = \{|b_i\rangle\}_{i=1}^d$. The probability of measuring a particular outcome "b_i" is $p_i = |\langle\Psi|b_i\rangle|^2$. After the measurement, if a certain outcome "b_i" is observed, then the state $|\Psi\rangle$ has collapsed to $|b_i\rangle$.

Pauli matrices

The Pauli matrices are 2×2 matrices,

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad Y = iXZ, \tag{128}$$

and the following relations hold:

$$X|0\rangle = |1\rangle, X|1\rangle = |0\rangle \qquad X|+\rangle = |+\rangle, X|-\rangle = -|-\rangle \tag{129}$$
$$Z|0\rangle = |0\rangle, Z|1\rangle = -|1\rangle \qquad Z|+\rangle = |-\rangle, Z|-\rangle = |+\rangle \tag{130}$$
$$Y|0\rangle = i|1\rangle, Y|1\rangle = -i|0\rangle \qquad Y|+\rangle = -i|-\rangle, Y|-\rangle = i|+\rangle \tag{131}$$

Bibliography

[1] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2000 (cited on page 2).

[2] B. Schumacher and M. Westmoreland. *Quantum Processes Systems, and Information*. Cambridge University Press, 2010 (cited on page 2).

Lecture Notes

Quantum Cryptography Week 1: Quantum tools and a first protocol

Contents

In this course you will learn about the basics of quantum communication and quantum cryptography. Unlike large scale quantum computers, both technologies are already implemented today. Yet it remains a grand challenge to do quantum communication and cryptography over long distances. This week we will learn about a very simple quantum protocol: we will encrypt quantum states! Yet, to prepare our entry into quantum communication and cryptography, we first need to learn a little more about quantum information. Even if you did not follow week 0, we recommend downloading the lecture notes for week 0 for notations and conventions used here.

1.1 Probability notation

Before we start we recall standard notation of classical probability theory which we use throughout these lecture notes. There are many good textbooks and online resources on probability theory available, such as [Kel94; Ros10], and we refer you to any of them for additional background.

Consider a discrete random variable X taking values in some alphabet $\mathscr{X}$ of size n. We write $P_X(\cdot)$ for the distribution of X, and $|X|$ for the size of the alphabet of X. The notation $P_X(x)$ denotes the probability that the random variable takes on a specific symbol $x \in \mathscr{X}$. When the distribution is clear from context, we use the shorthands $p_x = p(x) = P(X = x) = P_X(x)$. It will be useful to remember that a probability distribution $P_X(\cdot)$ is specified by non-negative probability values, i.e. $\forall x \in \mathscr{X}$, $P_X(x) \geq 0$. Furthermore, X should be normalized, which means $\sum_{x\in\mathscr{X}} P_X(x) = 1$.

■ **Example 1.1.1** Let $\mathscr{X} = \{1,2,3,4,5,6\}$ correspond to the faces of a 6 sided die. If the die is fair, i.e., all sides have equal probability of occuring then $P_X(x) = 1/6$ for all $x \in \mathscr{X}$. Using the shorthands, this reads $p_x = p(x) = 1/6$. The size of the alphabet is given by $|X| = 6$. ■

A random variable X can be correlated with another random variable Y. This means that they have a joint distribution, $P_{XY}(x,y)$, that is not necessarily a product, that is, $P_{XY}(x,y) \neq P_X(x)P_Y(y)$ in general. This leads to the notion of *conditional probabilities* $P_{X|Y}(x|y)$, where $P_{X|Y}(x|y)$ is the probability that X takes on the value x, conditional on the event that Y takes on the value y. As before, we will generally use the following shorthands when it is clear which random variable we refer to

$$p_{x|y} = p(x|y) = P(X = x|Y = y) = P_{X|Y}(x|y). \tag{1.1}$$

As you know from your probability class, Bayes rule relates this conditional probability to the joint probabilities. Since $P_{XY}(x,y) = P_X(x)P_{Y|X}(y|x) = P_Y(y)P_{X|Y}(x|y)$ we have

$$P_{X|Y}(x|y) = \frac{P_{XY}(x,y)}{P_Y(y)}\,, \tag{1.2}$$

whenever $P_Y(y) > 0$ [1].

■ **Example 1.1.2** Let's consider the fair die above, and an unfair die which always rolls a "6". That is, $\mathscr{X} = \{1,2,3,4,5,6\}$ in which $P_X(6) = 1$ and $P_X(x) = 0$ for $x \neq 6$. Let Y now refer to the choice of a fair, or unfair die. Suppose that we choose to roll the unfair or fair die with equal probability. That is $\mathscr{Y} = \{$"fair", "unfair"$\}$ where $P_Y(\text{fair}) = 1/2$ and $P_Y(\text{unfair}) = 1/2$. We thus have $P_{X|Y}(x|\text{fair}) = 1/6$ and $P_{X|Y}(6|\text{unfair}) = 1$ and $P_{X|Y}(x|\text{unfair}) = 0$ for $x \neq 6$. ■

Exercise 1.1.1 Compute the joint probability $P_{XY}(x,y)$ for the example of choosing a fair or unfair die. ■

[1] Note that the distribution over x given y is irrelevant if y cannot occur $P_Y(y) = 0$.

Exercise 1.1.2 Suppose now that we choose to roll the fair or unfair die with probability $P_Y(\text{fair}) = P_Y(\text{unfair}) = 1/2$, but don't tell you which one it is. However, I show you the outcome X of the die roll. That is, I have Y and you have X. Suppose that $X = 3$. What is the most likely die? I.e., is it more likely that $Y = \text{fair}$ or $Y = \text{unfair}$? How about $X = 6$? ■

1.2 Density matrices

Let us start by investigating a more general formalism for writing down quantum states. There are two motivations for doing so. Let's start with the basic question of how to write down the state of one of several qubits. To this end, imagine we have two quantum systems A and B. For example, A and B are two qubits in a joint state $|\psi\rangle_{AB}$ and we want to know the state of qubit A. If the joint state is $|\psi\rangle_{AB} = |\psi\rangle_A \otimes |\psi\rangle_B$, that is, it is obtained by taking the tensor product of qubit A in the state $|\psi\rangle_A$ and qubit B in the state $|\psi\rangle_B$, then the answer seems clear: A is simply in the state $|\psi\rangle_A$. However, if you took week 0, you may remember that some bipartite quantum states $|\psi\rangle_{AB}$, defined over two systems A and B, can be defined as superpositions of tensor products, in a way that makes it non-obvious whether the state can be directly written as a single tensor product. A good example of such a state is the EPR pair $|\text{EPR}\rangle_{AB} = \frac{1}{\sqrt{2}}|0\rangle_A|0\rangle_B + \frac{1}{\sqrt{2}}|1\rangle_A|1\rangle_B$. For such states we *cannot* express $|\psi\rangle_{AB} = |\psi\rangle_A \otimes |\psi\rangle_B$, that is as a tensor product of two states $|\psi\rangle_A$ on A and $|\psi\rangle_B$ on B. It is thus unclear how we could express the state of A without making any reference to B. Such a description should still be possible: after all, the state does exist! If it doesn't fit in our formalism of states as vectors it must mean the formalism is incomplete, and we need to find a mathematical generalization for it.

The second motivation for a more general description arises from a situation in which a probabilistic process, for example a measurement, prepares different states with some probability. Suppose we encounter a situation in which we had either a state $|\psi_1\rangle$ with some probability p_1, or a state $|\psi_2\rangle$ with probability p_2. To express the state accurately, we have to take into account both states and probabilities $\{|\psi_i\rangle, p_i\}_i$. Can we somehow write down the proper mathematical description of the state created by such a process?

1.2.1 Introduction

The answer to these questions lies in the so-called density matrix formalism. To start with, let us write down the quantum state $|\psi\rangle$ of a single system as a matrix $\rho = |\psi\rangle\langle\psi|$. Note that this is a rank-1 matrix, it has precisely 1 non-zero eigenvalue (equal to 1) with associated eigenstate $|\psi\rangle$.

■ **Example 1.2.1** Consider the following matrices corresponding to $|0\rangle$ and $|+\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$

$$|0\rangle\langle 0| = \begin{pmatrix} 1 \\ 0 \end{pmatrix} (1\ 0) = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \tag{1.3}$$

$$|+\rangle\langle +| = \frac{1}{2}\begin{pmatrix} 1 \\ 1 \end{pmatrix} (1\ 1) = \frac{1}{2}\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}. \tag{1.4}$$

■

How does writing down states as matrices help us to resolve the questions above? To see how, let us first consider the second motivation for a more general description. In particular, let us consider the case where someone prepares 2 possible states $|\psi_1\rangle$ and $|\psi_2\rangle$ with equal probability $p_1 = p_2 = 1/2$. Clearly, a superposition is not the correct description: The state really is in precisely one of the two states, with probability $1/2$ each. Indeed, the preparer knows the identity of the state. If the identity of the state is not known, however, how can we write down the resulting state? It turns out that we can describe the state of the resulting system as a *mixture* between $|\psi_1\rangle$ and $|\psi_2\rangle$.

For equal probabilities, this mixture becomes

$$\rho = \frac{1}{2}|\psi_1\rangle\langle\psi_1| + \frac{1}{2}|\psi_2\rangle\langle\psi_2| \,. \tag{1.5}$$

We also call such a ρ a *density matrix*. In general, if a source prepares the state $|\psi_x\rangle$ with probability p_x, the resulting system will be in the state

$$\rho = \sum_x p_x |\psi_x\rangle\langle\psi_x| \,. \tag{1.6}$$

Why would this be a good description? Let's consider what happens if we measure in the standard basis. If the system would actually be in the state $|\psi_j\rangle$, then we would expect the probabilities of outcomes to be

$$q_{0|j} = |\langle 0||\psi_j\rangle|^2 = \langle 0||\psi_j\rangle\langle\psi_j||0\rangle \,, \tag{1.7}$$
$$q_{1|j} = |\langle 1||\psi_j\rangle|^2 = \langle 1||\psi_j\rangle\langle\psi_j||1\rangle \,. \tag{1.8}$$

If state $|\psi_j\rangle$ is prepared with probability p_j, then we would expect the outcome probabilities to be

$$q_0 = \sum_j p_j q_{0|j} \,, \tag{1.9}$$
$$q_1 = \sum_j p_j q_{1|j} \,. \tag{1.10}$$

Let us expand one of these terms to relate to the density matrix formalism. We have

$$q_0 = \sum_j p_j q_{0|j} = \sum_j p_j \langle 0||\psi_j\rangle\langle\psi_j||0\rangle = \langle 0| \left(\sum_j p_j |\psi_j\rangle\langle\psi_j| \right) |0\rangle = \langle 0|\rho|0\rangle \,. \tag{1.11}$$

The density matrix ρ thus accurately reflects what we would intuitively expect from the probabilities of measurement outcomes.

■ **Example 1.2.2** If a source prepares quantum states in a probabilistic manner, i.e. it prepares the quantum state ρ_x with probability p_x, then the resulting *density matrix* is given by

$$\rho = \sum_x p_x \rho_x \,. \tag{1.12}$$

The set of probabilities and density matrices $\mathscr{E} = \{(p_x, \rho_x)\}_x$ is also called an *ensemble* of states. Note that the case where the source prepares pure states is a special case with $\rho_x = |\psi_x\rangle\langle\psi_x|$ and $p_x = 1$ for a single x. ■

■ **Example 1.2.3** Suppose the source prepares $|0\rangle\langle 0|$ with probability $1/2$, and $|+\rangle\langle +|$ with probability $1/2$. Then the resulting density matrix for the ensemble $\{(1/2, |0\rangle\langle 0|), (1/2, |+\rangle\langle +|)\}$ is given by

$$\rho = \frac{1}{2}|0\rangle\langle 0| + \frac{1}{2}|+\rangle\langle +| = \frac{1}{2}\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} + \frac{1}{4}\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} = \frac{1}{4}\begin{pmatrix} 3 & 1 \\ 1 & 1 \end{pmatrix} \,. \tag{1.13}$$

■

■ **Example 1.2.4** Superposition is not the same as a mixture. Intuitively, the difference is that a mixture is an inherently classical mixture: there is a process that prepares one *or* the other with some probability. In contrast, a state in a superposition is one *and* the other. To see the difference,

let us consider mixing or creating a superposition of $|0\rangle$ and $|1\rangle$. Consider a source that prepares the state $|0\rangle$ and $|1\rangle$ with probabilities $p_0 = p_1 = 1/2$. Suppose we measure the resulting state

$$\rho = \frac{1}{2}|0\rangle\langle 0| + \frac{1}{2}|1\rangle\langle 1| = \mathbb{I}/2\,, \tag{1.14}$$

where

$$\mathbb{I} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}\,. \tag{1.15}$$

in the Hadamard basis $\{|+\rangle, |-\rangle\}$ with

$$|+\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right)\,, \tag{1.16}$$

$$|-\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right)\,. \tag{1.17}$$

We have that the probabilities of outcomes are given by

$$q_+ = \langle +|\rho|+\rangle = \frac{1}{2}\,, \tag{1.18}$$

$$q_- = \langle -|\rho|-\rangle = \frac{1}{2}\,. \tag{1.19}$$

In contrast, consider now the superposition $|+\rangle$. Measuring $|+\rangle$ in the Hadamard basis, results in $q_+ = 1$ and $q_- = 0$. This illustrates a fundamental difference between mixtures and superpositions. ■

Exercise 1.2.1 If $|\Psi\rangle$ is an n-qubit quantum state, what are the dimensions of the density matrix $|\Psi\rangle\langle\Psi|$? ■

R It is crucial to note that unlike in the case of classical probability distributions, the same density matrix can be obtained from different ensembles. A simple example is provided by the operator

$$\rho = \frac{\mathbb{I}}{2}\,, \tag{1.20}$$

which is also called the *maximally mixed* state. You may verify that

$$\frac{\mathbb{I}}{2} = \frac{1}{2}\left(|0\rangle\langle 0| + |1\rangle\langle 1|\right) = \frac{1}{2}\left(|+\rangle\langle +| + |-\rangle\langle -|\right)\,. \tag{1.21}$$

1.2.2 Some mathematical definitions

To formally define density matrices and their properties, we recall some important notions from linear algebra. The first term we introduce is the *linear operator*, which is, in our context, just a fancy name to express matrices. Such a term highlights the idea that a matrix maps one vector to another. It can hence be thought of as an operation performed on vectors, which - since matrix multiplication is linear - will be a linear operation. We will hence use matrix and operator interchangeably. To make sense of the quantum literature, however, the following definitions will be useful.

Definition 1.2.1 — Linear operator. Consider a d-dimensional complex vector space $\mathbb{C}^d$. A *linear* operator $L : \mathbb{C}^d \to \mathbb{C}^{d'}$ can be represented as a $d' \times d$ matrix,

$$L = \begin{pmatrix} L_{11} & L_{12} & \cdots & L_{1d} \\ L_{21} & \ddots & \ddots & L_{2d} \\ \vdots & \ddots & \ddots & \vdots \\ L_{d'1} & L_{d'2} & \cdots & L_{d'd} \end{pmatrix}, \tag{1.22}$$

where each element $L_{ij} \in \mathbb{C}$. The set of linear operators is denoted $\mathscr{L}(\mathbb{C}^d, \mathbb{C}^{d'})$.

Definition 1.2.2 — Hermitian matrix M. A linear operator $M \in \mathscr{L}(\mathbb{C}^d, \mathbb{C}^d)$ is *Hermitian* if $M^\dagger = M$.

The spectral theorem states that any Hermitian operator $M \in \mathscr{L}(\mathbb{C}^d, \mathbb{C}^d)$ can be diagonalized with real eigenvalues. This means that there exists an orthonormal basis $\{|v_j\rangle\}$ of $\mathbb{C}^d$ (the *eigenvectors*) and real numbers λ_j (the *eigenvalues*) such that $M = \sum_j \lambda_j |v_j\rangle\langle v_j|$.

Definition 1.2.3 — Positive semidefinite matrix. A Hermitian matrix M is *positive semidefinite* if all its eigenvalues $\{\lambda_i\}_i$ are non-negative, i.e. $\lambda_i \geq 0$. This condition is often denoted as $M \geq 0$.

Exercise 1.2.2 Show that a matrix M is positive semidefinite if and only if $\langle v|M|v\rangle \geq 0$ for all unit vectors $|v\rangle$. In particular, the diagonal coefficients $\langle i|M|i\rangle$ of M in any basis are non-negative. Show that this is not a sufficient condition: find an M such that the diagonal coefficients of M are all positive but M itself is not positive semidefinite. ■

An important operation on matrices is the *trace*. We already saw in Week 0 that we can express it simply as the sum of the diagonal elements. As such, the trace is a linear map which takes any matrix to a complex number. It will sometimes be convenient to note that the trace can also be expressed as follows:

Definition 1.2.4 — Trace of a matrix. The trace of a matrix $M \in \mathscr{L}(\mathbb{C}^d, \mathbb{C}^d)$ is defined as

$$\mathrm{tr}(M) = \sum_i \langle i|M|i\rangle,$$

where $\{|i\rangle\}$ is any orthonormal basis of $\mathbb{C}^d$.

You should convince yourself that the definition of the trace does not depend on the choice of orthonormal basis! An important property of the trace is that it is *cyclic*:

Exercise 1.2.3 Show that for any matrices M, N we have $\mathrm{tr}(MN) = \mathrm{tr}(NM)$. We will often use this property to perform manipulations such as

$$\langle i|M|i\rangle = \mathrm{tr}(\langle i|M|i\rangle) = \mathrm{tr}(M|i\rangle\langle i|). \tag{1.23}$$

It is however worth noting that in general, a non-cyclic permutation of the matrices do not preserve the trace. More precisely, for matrices M, N, P, in general

$$\mathrm{tr}(MNP) \neq \mathrm{tr}(NMP). \tag{1.24}$$

■

1.2.3 Density matrices and their properties

Given the discussion above we are motivated to take the density matrix ρ as a more general description of quantum states. Before was can make this formally precise, let us first investigate when some matrix ρ would actually be considered a valid density matrix, that is, a description of a quantum state. It turns out that there are two necessary (and sufficient) properties in order for a density matrix to represent a valid quantum state: it should be *positive semidefinite* and have *trace equal to 1*. To see why this is true, consider the diagonalized representation of a density matrix ρ into its eigenvalues $\{\lambda_j\}_j$ and corresponding eigenvectors $\{|v_j\rangle\}_j$ as

$$\rho = \sum_j \lambda_j |v_j\rangle\langle v_j| \tag{1.25}$$

where the vectors $|v_j\rangle$ are orthonormal. Let us imagine that we measure ρ in some other orthonormal basis $\{|w_k\rangle\}_k$. Thinking about a process that prepares a certain state $|v_j\rangle\langle v_j|$ with probability λ_j, we could imagine that we measure just $|v_j\rangle$ in that basis. We know that in this case, the probability of obtaining measurement outcome k (conditioned on the preparation being in state $|v_j\rangle\langle v_j|$) is given by

$$q_{k|j} = |\langle v_j|w_k\rangle|^2 = \langle w_k||v_j\rangle\langle v_j||w_k\rangle \ . \tag{1.26}$$

Hence the probability of obtaining outcome k when measuring ρ should be given by

$$q_k = \sum_j \lambda_j q_{k|j} = \langle w_k| \left(\sum_j \lambda_j |v_j\rangle\langle v_j| \right) |w_k\rangle = \langle w_k|\rho|w_k\rangle \ . \tag{1.27}$$

Note that we must have $q_k \geq 0$ and $\sum_k q_k = 1$. By imagining that we measure ρ in its eigenbasis, that is, $|w_j\rangle = |v_j\rangle$, it is easy to see that $\lambda_j \geq 0$, that is, ρ is a *positive semidefinite* matrix. We also have $\mathrm{tr}(\rho) = 1$, since

$$1 = \sum_j q_j = \sum_j \lambda_j \mathrm{tr}(|v_j\rangle\langle v_j|) = \mathrm{tr}(\rho) \ . \tag{1.28}$$

This motivates the following definition of a density matrix, which is the most general way to describe the state of a quantum system.

Definition 1.2.5 — Density matrix. Consider a quantum system with state space $\mathbb{C}^d$. A *density matrix*, commonly denoted as ρ, is a linear operator $\rho \in \mathscr{L}(\mathbb{C}^d, \mathbb{C}^d)$ such that:
1. $\rho \geq 0$, and
2. $\mathrm{tr}(\rho) = 1$.
If $\mathrm{rank}(\rho) = 1$, then ρ is called a *pure* state, otherwise ρ is *mixed*.

Let us also summarize the rule for computing outcome probabilities for measuring a quantum system described by the density matrix ρ motivated by our discussions.

Definition 1.2.6 — Measuring a density matrix in a basis. Consider a quantum system in the state ρ. Measuring ρ in the basis $\{|b_j\rangle\}_j$ results in outcome j with probability

$$q_j = \langle b_j|\rho|b_j\rangle \ . \tag{1.29}$$

1.2.4 Bloch representation for one qubit mixed states

In week 0, we saw that one qubit states have a nice graphical representation in terms of vectors on the Bloch sphere. In particular, any pure quantum state can be described by a *Bloch vector*

$\vec{r} = (\cos\phi \sin\theta, \sin\phi \sin\theta, \cos\theta)$. Is this the same for mixed states? The answer to this turns out to be yes! Concretely, we can write any one qubit density matrix as

$$\rho = \frac{1}{2}(\mathbb{I} + v_x X + v_z Z + v_y Y) \ , \tag{1.30}$$

where X, Y, Z are the Pauli matrices you have encountered in Week 0, and if ρ is pure, then the vector $\vec{v} = (v_x, v_y, v_z)$ is precisely the Bloch vector $\vec{r}$ that you already know! For pure states $\|\vec{v}\|_2^2 = v_x^2 + v_y^2 + v_y^2 = 1$. For mixed states, however, we can have $\|\vec{v}\|_2^2 \leq 1$. Thus for the case of 2×2 matrices the vector $\vec{v}$ tells us immediately whether the matrix ρ is a valid one qubit quantum state! This is the case if and only if $\|\vec{v}\|_2 \leq 1$.

Note that the matrices $\mathscr{S} = \{\mathbb{I}, X, Z, Y\}$ form a basis for the space of 2×2 density matrices that correspond to a qubit. You should convince yourself that all these matrices are orthogonal under the Hilbert-Schmidt inner product $\langle A, B\rangle = \mathrm{tr}(A^\dagger B)$. That is,

$$\mathrm{tr}(X^\dagger Y) = \mathrm{tr}(X^\dagger Z) = \mathrm{tr}(X^\dagger \mathbb{I}) = 0 \ , \tag{1.31}$$

and similarly for all other pairs of matrices.

Exercise 1.2.4 Using the orthogonality condition (1.31), show that

$$|0\rangle\langle 0| = \frac{1}{2}(\mathbb{I} + Z) \ , \tag{1.32}$$

$$|1\rangle\langle 1| = \frac{1}{2}(\mathbb{I} - Z) \ , \tag{1.33}$$

■

Exercise 1.2.5 Use the fact that all matrices $M, N \in \mathscr{S}$ with $M \neq N$ anti-commute, i.e., $\{M, N\} = MN + NM = 0$ to show that $\mathrm{tr}(MN) = 0$ whenever $M \neq N \in \mathscr{S}$. ■

1.3 Combining density matrices

If we have two quantum systems A and B, described by density matrices ρ_A and ρ_B, how can we write down the joint state ρ_{AB}? We saw in Week 0 that two pure quantum states which can be represented by vectors $|v_1\rangle \in \mathbb{C}^{d_1}, |v_2\rangle \in \mathbb{C}^{d_2}$ can be combined by taking their tensor product $|v_1\rangle \otimes |v_2\rangle \in \mathbb{C}^{d_1} \otimes \mathbb{C}^{d_2}$. It turns out that the rule for mixed states is very similar, and a simple extension of the concept of the tensor product. Let us start with the simple case where ρ_A, ρ_B are 2×2-dimensional matrices,

$$\rho_A \otimes \rho_B = \begin{pmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{pmatrix} \otimes \begin{pmatrix} n_{11} & n_{12} \\ n_{21} & n_{22} \end{pmatrix} = \begin{pmatrix} m_{11}\begin{pmatrix} n_{11} & n_{12} \\ n_{21} & n_{22} \end{pmatrix} & m_{12}\begin{pmatrix} n_{11} & n_{12} \\ n_{21} & n_{22} \end{pmatrix} \\ m_{21}\begin{pmatrix} n_{11} & n_{12} \\ n_{21} & n_{22} \end{pmatrix} & m_{22}\begin{pmatrix} n_{11} & n_{12} \\ n_{21} & n_{22} \end{pmatrix} \end{pmatrix} \tag{1.34}$$

$$= \begin{pmatrix} m_{11}n_{11} & m_{11}n_{12} & m_{12}n_{11} & m_{12}n_{12} \\ m_{11}n_{21} & m_{11}n_{22} & m_{12}n_{21} & m_{12}n_{22} \\ m_{21}n_{11} & m_{21}n_{12} & m_{22}n_{11} & m_{22}n_{12} \\ m_{21}n_{21} & m_{21}n_{22} & m_{22}n_{21} & m_{22}n_{22} \end{pmatrix} . \tag{1.35}$$

For example, if we have two density matrices $\rho_A = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$ and $\rho_B = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$, then

$$\rho_{AB} = \rho_A \otimes \rho_B = \begin{pmatrix} 1\cdot\rho_B & 0\cdot\rho_B \\ 0\cdot\rho_B & 0\cdot\rho_B \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}. \tag{1.36}$$

This definition easily extends to larger matrices as follows:

Definition 1.3.1 — Tensor product. Consider any $d' \times d$ matrix ρ_A and $k' \times k$ matrix ρ_B,

$$\rho_A = \begin{pmatrix} m_{11} & m_{12} & \cdots & m_{1d} \\ m_{21} & \ddots & \ddots & m_{2d} \\ \vdots & \ddots & \ddots & \vdots \\ m_{d'1} & m_{d'2} & \cdots & m_{d'd} \end{pmatrix}, \qquad \rho_B = \begin{pmatrix} n_{11} & n_{12} & \cdots & n_{1k} \\ n_{21} & \ddots & \ddots & n_{2k} \\ \vdots & \ddots & \ddots & \vdots \\ n_{k'1} & n_{k'2} & \cdots & n_{k'k} \end{pmatrix}. \tag{1.37}$$

The tensor product of these matrices is given by a $d'k' \times dk$ matrix,

$$\rho_{AB} = \rho_A \otimes \rho_B = \begin{pmatrix} m_{11}B & m_{12}B & \cdots & m_{1d}B \\ m_{21}B & \ddots & \ddots & m_{2d}B \\ \vdots & \ddots & \ddots & \vdots \\ m_{d'1}B & m_{d'2}B & \cdots & m_{d'd}B \end{pmatrix}. \tag{1.38}$$

As a word of caution, we note that the tensor product, like the usual matrix product, is non-commutative.

■ **Example 1.3.1** Consider the density matrices $\rho_A = \frac{1}{4}\begin{pmatrix} 1 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 1 \end{pmatrix}$ and $\rho_B = \frac{1}{2}\begin{pmatrix} 1 & -i \\ i & 1 \end{pmatrix}$. Then

$$\rho_A \otimes \rho_B = \frac{1}{8}\begin{pmatrix} 1 & -i & 1 & -i & 0 & 0 \\ i & 1 & i & 1 & 0 & 0 \\ 1 & -i & 2 & -2i & 1 & -i \\ i & 1 & 2i & 2 & i & 1 \\ 0 & 0 & 1 & -i & 1 & -i \\ 0 & 0 & i & 1 & i & 1 \end{pmatrix}, \tag{1.39}$$

and

$$\rho_B \otimes \rho_A = \frac{1}{8}\begin{pmatrix} 1 & 1 & 0 & -i & -i & 0 \\ 1 & 2 & 1 & -i & -2i & -i \\ 0 & 1 & 1 & 0 & -i & -i \\ i & i & 0 & 1 & 1 & 0 \\ i & 2i & i & 1 & 2 & 1 \\ 0 & i & i & 0 & 1 & 1 \end{pmatrix} \neq \rho_A \otimes \rho_B. \tag{1.40}$$

■

1.4 Classical-quantum states

Throughout quantum cryptography, we often find ourselves in a situation in which the honest parties have some classical information X about which an adversary - like an eavesdropper Eve - may hold some quantum information Q. It is worth thinking about that the joint states ρ_{XQ} have a very special structure.

1.4.1 Classical states

As a first step, let us first pause to think about what it means that X is "classical information". To this end, it is interesting to note that it is possible to write a probability distribution over classical strings x in terms of density matrices. Suppose that we have a classical probability distribution over symbols from the alphabet $\mathscr{X} = \{0,\dots,d-1\}$, where p_x denotes the probability of observing symbol x. Identifying classical bits (or indeed numbers) with elements of the standard basis $\{|0\rangle,\dots,|d-1\rangle\}$, we can express a source preparing each of the possible states $|x\rangle$ with probability p_x by the mixture

$$\rho = \sum_{x=0}^{d-1} p_x |x\rangle\langle x| \,. \tag{1.41}$$

Note that ρ is a density matrix which has the probabilities p_x on the diagonal and is otherwise zero. As such, ρ is just another way to express the probability distribution p_x. Indeed, you may want to check that measuring ρ in the standard basis results precisely in obtaining outcome "x" with probability p_x.

Definition 1.4.1 — Classical state. Consider a system X with state space $\mathbb{C}^d$, and let $\{|x\rangle\}_{x=0}^{d-1}$ denote the standard basis for $\mathbb{C}^d$. A system X is in a classical state, or *c-state*, when the corresponding density matrix ρ_X is diagonal in the standard basis of the state space of X, i.e. ρ_X has the form

$$\rho = \sum_{x=0}^{d-1} p_x |x\rangle\langle x| \tag{1.42}$$

where $\{p_x\}_{x=0}^{d-1}$ is any normalized probability distribution.

In quantum cryptography, we will often encounter states which are partially classical, and partially quantum. Suppose we prepare the following states for Alice and Bob. With probability $1/2$ we prepare $|0\rangle\langle 0|_X \otimes \rho_0^Q$ with $\rho_0^Q = \frac{\mathbb{I}_Q}{2}$, and with probability $1/2$ we prepare $|1\rangle\langle 1|_X \otimes \rho_1^Q$ with $\rho_1^Q = |+\rangle\langle +|$. The joint state is a *classical quantum state*, or cq-state of the form

$$\rho_{XQ} = \frac{1}{2} \sum_{x\in\{0,1\}} |x\rangle\langle x|_X \otimes \rho_x^Q \,. \tag{1.43}$$

Note that in this case Alice knows which state Bob is given. However, as we will see later, Bob cannot learn which x Alice holds with certainty. (Intuitively, the reason is that, while Alice's states are orthonormal, Bob's states have "overlap" and are not perfectly distinguishable.)

Definition 1.4.2 A classical-quantum state, or simply called a *cq-state* takes the form

$$\rho_{XQ} = \sum_x p_x |x\rangle\langle x|_X \otimes \rho_x^Q \,. \tag{1.44}$$

That is, it consists of a classical register X and a quantum register Q. If Q is absent, then ρ_X is simply a classical state.

In applications to cryptography x will often represent some (partially secret) classical string that Alice creates during a quantum protocol, and ρ_x^Q some quantum information that an attacker may have gathered during the protocol, and which may be correlated with the string x. It is an established custom in the quantum information literature to use letters X, Y, Z to denote such classical registers, and reserve the other letters for quantum information.

1.5 General measurements

So far we have only been measuring quantum states in a given basis. Quantum mechanics allows a much more refined notion of measurement, which plays an important role both in quantum information theory and cryptography. On the one hand, in quantum information theory certain tasks, such as the task of discriminating between multiple states, can be solved more efficiently using these generalized measurements. On the other hand, taking an adversarial viewpoint, in quantum cryptography it is essential that we prove security for the most general kind of attack, including all measurements that an attacker could possibly make!

1.5.1 POVMs

If we are only interested in the probabilities of measurement outcomes - but not what happens after the measurement - then the most general kind of measurement that is allowed in quantum mechanics can be described by a positive operator-valued measure, or POVM for short. It can be defined as follows.

Definition 1.5.1 — POVM. A POVM on $\mathbb{C}^d$ is a set of positive semidefinite operators $\{M_x\}_{x\in\mathscr{X}}$ such that

$$\sum_x M_x = \mathbb{I}_{\mathbb{C}^d} \ . \tag{1.45}$$

The subscript x is used as a label for the measurement outcome. The probability p_x of observing outcome x can be expressed using the *Born rule* as

$$p_x = \mathrm{tr}(M_x\rho) \ . \tag{1.46}$$

■ **Example 1.5.1** Recall that when measuring a state $|\psi\rangle = \sum_x \alpha_x|x\rangle$ in a basis such as $\{|x\rangle\}_x$, the probability of outcome x is simply given by $|\alpha_x|^2$. Let's see how this can be formulated as a special case of the POVM formalism we just introduced. For each x let $M_x = |x\rangle\langle x|$, so that M_x is positive semidefinite (it fact, it is a projector, i.e. $M^2 = M$) and $\sum_x M_x = \mathbb{I}$ ($\{|x\rangle\}$ is a basis), as required. We can then use the Born rule to compute

$$\begin{aligned} p_x &= \mathrm{tr}(M_x\rho) \\ &= \mathrm{tr}(|x\rangle\langle x|\rho) \\ &= \langle x|\rho|x\rangle \\ &= \sum_{x',x''} \alpha_{x'}\alpha^*_{x''}\langle x|x'\rangle\langle x''|x\rangle \\ &= |\alpha_x|^2 \ . \end{aligned}$$

■

■ **Example 1.5.2** Consider a distribution (p_x) and the classical mixture $\rho = \sum_x p_x|x\rangle\langle x|$. If we measure ρ in the standard basis, with associated POVM $M_x = |x\rangle\langle x|$ as in the previous example, we obtain outcome x with probability

$$\mathrm{tr}(|x\rangle\langle x|\rho) = \langle x|\rho|x\rangle = p_x. \tag{1.47}$$

Thus ρ indeed captures the classical distribution given by the probabilities p_x. ■

You may wonder what happens to a quantum state after a generalized measurement has been performed. For the case of measuring in a basis, the answer is simple: the state collapses to the basis element associated with the outcome of the measurement that is obtained.

In the case of a POVM it turns out that the information given by the operators $\{M_x\}$ is not sufficient to fully determine the post-measurement state. Intuitively the reason is because such a measurement may not fully collapse the state (the post-measurement state may not be pure), and as a consequence there remains the flexibility to apply an arbitrary unitary on the post-measurement state, without affecting the outcome probabilities.

In order to specify post-measurement states we need to give a *Kraus operator representation* of the POVM.

Definition 1.5.2 — Kraus operators. Let $M = \{M_x\}$ be a given POVM on $\mathbb{C}^d$. A *Kraus operator representation* of M is a set of linear operators $A_x \in \mathscr{L}(\mathbb{C}^d, \mathbb{C}^{d'})$ such that $M_x = A_x^\dagger A_x$ for all x.

Note that a Kraus decomposition of any POVM always exists by setting $A_x = \sqrt{M_x}$, the positive square root of M_x. (For any positive semidefinite matrix N, if $N = \sum_i \lambda_i |v_i\rangle\langle v_i|$ is the spectral decomposition of N, then $\sqrt{N} = \sum_i \sqrt{\lambda_i}|v_i\rangle\langle v_i|$.) In particular, if $M_x = |u_x\rangle\langle u_x|$ is a projector then $\sqrt{M_x} = M_x$ and we can take $A_x = M_x$. But for any unitary U_x on $\mathbb{C}^d$, $A'_x = U_x\sqrt{M_x}$ is also a valid decomposition. Hence, there is no unique Kraus representation for a given POVM.

1.5.2 Generalized measurements

The most general form to write down a quantum measurement is thus given by the full set of Kraus operators $\{A_x\}_x$. From these, we can easily find the POVM operators as $M_x = A_x^\dagger A_x$, but also compute the post-measurement states.

Definition 1.5.3 — Post-measurement state. Let ρ be a density matrix and $M = \{M_x\}$ a POVM with Kraus decomposition given by operators $\{A_x\}$. Suppose the measurement is preformed and the outcome x is obtained. Then the state of the system after the measurement, conditioned on the outcome x, is

$$\rho_{|x} = \frac{A_x \rho A_x^\dagger}{\mathrm{tr}(A_x^\dagger A_x \rho)} .$$

You may want to convince yourself that when measuring a pure state $|\psi\rangle$ in the standard basis, with POVM elements $M_x = |x\rangle\langle x|$ and Kraus decomposition $A_x = M_x = |x\rangle\langle x|$, the post-measurement state as defined above is precisely the basis state associated to the measurement outcome. Note that since a POVM does not have a unique decomposition into Kraus operators, specifying POVM operators alone is insufficient to determine the post-measurement state. Nevertheless, talking about a POVM is extremely useful if we only care about measurement probabilities, since the matrices M_x have a slightly simpler form. In particular, we will note later, we can easily optimize over them using a semidefinite program (SDP).

An important class of generalized measurements is given by the case where the M_x are *projectors* onto orthogonal subspaces.

Definition 1.5.4 A *projective measurement*, also called a *von Neumann measurement*, is given by a set of orthogonal projectors $M_x = \Pi_x$ such that $\sum_x \Pi_x = \mathbb{I}$. For such a measurement, unless otherwise specified we will always use the default Kraus decomposition $A_x = \Pi_x$. The probability q_x of observing measurement outcome x can then be expressed as

$$q_x = \mathrm{tr}(\Pi_x \rho),$$

and the post-measurement states are

$$\rho_{|x} = \frac{\Pi_x \rho \Pi_x}{\mathrm{tr}(\Pi_x \rho)} .$$

■ **Example 1.5.3** Suppose given a two-qubit state ρ, such that we would like to measure the parity (in the standard basis) of the two qubits. A first way to do this would be to measure ρ in the standard basis, obtain two bits, and take their parity. In this case the probability of obtaining the outcome "even" would be

$$q_{\text{even}} = \langle 00|\rho|00\rangle + \langle 11|\rho|11\rangle,$$

and the post-measurement state would be the mixture of the two post-measurement states associated with outcomes $(0,0)$ and $(1,1)$, so

$$\rho_{|\text{even}} = \langle 00|\rho|00\rangle|00\rangle\langle 00| + \langle 11|\rho|11\rangle|11\rangle\langle 11|.$$

Now suppose we measure the parity using a generalized measurement which directly projects onto the relevant subspaces, without measuring the qubits individually. That is, consider the projective measurement $\Pi_{\text{even}} = |00\rangle\langle 00| + |11\rangle\langle 11|$ and $\Pi_{\text{odd}} = \mathbb{I} - \Pi_{\text{even}} = |01\rangle\langle 01| + |10\rangle\langle 10|$. With this measurement the probability of obtaining the outcome "even" is

$$q'_{\text{even}} = \text{tr}(\Pi_{\text{even}}\rho) = \langle 00|\rho|00\rangle + \langle 11|\rho|11\rangle \ , \tag{1.48}$$

as before. However, the post-measurement state is now

$$\rho'_{|\text{even}} = \Pi_{\text{even}}\rho\Pi_{\text{even}} \ . \tag{1.49}$$

To see the difference, consider the state $\rho = |\text{EPR}\rangle\langle\text{EPR}|$ where $|\text{EPR}\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$. Then clearly the measurement should report the outcome "even" with probability 1, and you can check this is the case for both measurements. However, the post-measurement states are different. In the first case,

$$\rho_{|\text{even}} = \frac{1}{2}|00\rangle\langle 00| + \frac{1}{2}|11\rangle\langle 11|,$$

while in the second case,

$$\rho'_{|\text{even}} = |\text{EPR}\rangle\langle\text{EPR}|$$

is unchanged! This is one of the main advantages of using generalized measurements as opposed to basis measurements: they allow to compute certain simple quantities on multi-qubit states (such as the parity) without fully "destroying" the state, as happens when measuring in a basis. ■

Exercise 1.5.1 Use a projective measurement to measure the parity, in the Hadamard basis, of the state $|00\rangle\langle 00|$. Compute the probabilities of obtaining measurement outcomes "even" and "odd", and the resulting post-measurement states. What would the post-measurement states have been if you had first measured the qubits individually in the Hadamard basis, and then taken the parity? ■

1.6 The partial trace

Going back to our initial motivation for introducing density matrices, let's now give an answer to the following question: given a multi-qubit state, how do we write down the "partial state" associated to a subset of the qubits? More generally, suppose ρ_{AB} is a density matrix on a tensor product space $\mathbb{C}_A^{d_A} \otimes \mathbb{C}_B^{d_B}$, but suppose Alice holds the part of ρ corresponding to system A and Bob holds the part corresponding to system B. How do we describe the state ρ_A of Alice's system?

1.6.1 An operational viewpoint

The operation that takes us from ρ_{AB} to ρ_A is called the *partial trace*. It can be given a purely mathematical description that we will give below. However, before that, let's try to think about the problem from an operational point of view. First, an easy case: if $\rho_{AB} = \rho_A \otimes \rho_B$, where ρ_A and ρ_B are both density matrices, then clearly Alice's system is defined by ρ_A. A slightly more complicated case would be when $\rho_{AB} = \sum_i p_i \rho_i^A \otimes \rho_i^B$ is a mixture of tensor products (we will later see this is called a "separable state"); in this case it would certainly be natural to say that Alice's state is ρ_i^A with probability p_i, i.e. $\rho_A = \sum_i p_i \rho_i^A$.

How do we deal with a general ρ? The idea is to *imagine* that Bob performs a complete basis measurement on his system, using an arbitrary basis $\{|u_x\rangle\}$. Let's introduce a POVM on the joint system of Alice and Bob that models this measurement: since Alice does nothing, we can set $M_x = \mathbb{I}_A \otimes |u_x\rangle\langle u_x|_B$, which you can check indeed defines a valid POVM. Moreover, this is a projective measurement, so we can take the Kraus operators $A_x = M_x$ as well. By definition the post-measurement states are given by

$$\rho_{|x}^{AB} = \frac{M_x \rho_{AB} M_x}{\mathrm{Tr}\left(M_x \rho_{AB}\right)} = \frac{\left((\mathbb{I}_A \otimes \langle u_x|)\rho_{AB}(\mathbb{I}_A \otimes |u_x\rangle)\right)_A \otimes |u_x\rangle\langle u_x|_B}{\mathrm{Tr}\left((\mathbb{I}_A \otimes |u_x\rangle\langle u_x|_B)\rho_{AB}\right)}.$$

Notice how we wrote the state, as a tensor product of a state on A and one on B. Make sure you understand the notation in this formula.

Now the key step is to realize that, whatever the state of Alice's system A is, it shouldn't depend on any operation that Bob performs on B. After all, it may be that A is here on earth, and B on Mars and even quantum mechanics does not allow faster than light communication. As long as the two of them remain perfectly isolated, meaning that Alice doesn't get to learn the measurement that Bob performs or its outcome, then her state is unchanged. We can thus describe it as "with probability $q_x = \mathrm{Tr}(M_x \rho_{AB})$, Alice's state is the A part of $\rho_{|x}^{AB}$, i.e.

$$\rho_A = \sum_x q_x \frac{\left((\mathbb{I} \otimes \langle u_x|)\rho_{AB}(\mathbb{I} \otimes |u_x\rangle)\right)_A}{\mathrm{Tr}\left((\mathbb{I} \otimes |x\rangle\langle x|)\rho_{AB}\right)} = \sum_x (\mathbb{I} \otimes \langle u_x|)\rho_{AB}(\mathbb{I} \otimes |u_x\rangle). \tag{1.50}$$

Although we derived the above expression for Alice's state using sensible arguments, there is something you should be worried about: doesn't it depend on the choice of basis $\{|u_x\rangle\}$ we made for Bob's measurement? Of course, it should not, as our whole argument is based on the idea that Alice's reduced state should not depend on any operation performed by Bob. (We emphasize that this is only the case as long as Alice doesn't learn the measurement outcome! If we fix a particular outcome x then it's a completely different story; beware of the subtlety.)

Exercise 1.6.1 Verify that the state ρ_A defined in Eq.(1.50) does not depend on the choice of basis $\{|u_x\rangle\}$. [Hint: first argue that if two density matrices ρ, σ satisfy $\langle\phi|\rho|\phi\rangle = \langle\phi|\sigma|\phi\rangle$ for all unit vectors $|\phi\rangle$ then $\rho = \sigma$. Then compute $\langle\phi|\rho_A|\phi\rangle$, and use the POVM condition $\sum_x M_x = \mathbb{I}$ to check that you can get an expression independent of the $\{|u_x\rangle\}$. Conclude that ρ_A itself does not depend on $\{|u_x\rangle\}$.] ■

■ **Example 1.6.1** Consider the example of the EPR pair

$$|\mathrm{EPR}\rangle_{AB} = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \tag{1.51}$$

Writing this as a density operator we have

$$\rho_{AB} = |\mathrm{EPR}\rangle\langle\mathrm{EPR}|_{AB} = \frac{1}{2}\left(|00\rangle\langle 00| + |00\rangle\langle 11| + |11\rangle\langle 00| + |11\rangle\langle 11|\right). \tag{1.52}$$

Let's measure system B in the standard basis: taking A into account we consider the POVM $M_0 = \mathbb{I}_A \otimes |0\rangle\langle 0|_B$ and $M_1 = \mathbb{I}_A \otimes |1\rangle\langle 1|_B$. We can then compute

$$\begin{aligned} q_0 &= \mathrm{Tr}(M_0\rho) \\ &= \frac{1}{2}\mathrm{Tr}\left((\mathbb{I}\otimes|0\rangle\langle 0|)(|00\rangle\langle 00| + |00\rangle\langle 11| + |11\rangle\langle 00| + |11\rangle\langle 11|)\right) \\ &= \frac{1}{2}(1+0+0+0) = \frac{1}{2}, \end{aligned}$$

and similarly $q_1 = 1/2$. The post-measurement stated on A is then

$$\rho^A_{|0} = \frac{1}{2}(\mathbb{I}\otimes\langle 0|)\rho_{AB}(\mathbb{I}\otimes|0\rangle) + \frac{1}{2}(\mathbb{I}\otimes\langle 1|)\rho_{AB}(\mathbb{I}\otimes|1\rangle) = \frac{1}{2}|0\rangle\langle 0| + \frac{1}{2}|1\rangle\langle 1|.$$

Exercise: do the same calculation using a measurement in the Hadamard basis on B, and check that you get the same result! ∎

1.6.2 A mathematical definition

Armed with our "operational" definition of what the partial trace *should* be, we can now give the precise, mathematical definition of the partial trace operation.

Definition 1.6.1 — Partial Trace. Consider a general state

$$\rho_{AB} = \sum_{ijk\ell} \gamma_{ij}^{k\ell} |i\rangle\langle j|_A \otimes |k\rangle\langle \ell|_B, \tag{1.53}$$

where $|i\rangle_A, |j\rangle_A$ and $|k\rangle_B, |\ell\rangle_B$ run over orthonormal bases of A and B respectively. Then the partial trace over B is defined as

$$\rho_A = \mathrm{tr}_B(\rho_{AB}) = \sum_{ijk\ell} \gamma_{ij}^{k\ell} |i\rangle\langle j| \otimes \mathrm{tr}(|k\rangle\langle \ell|) = \sum_{ij}\left(\sum_k \gamma_{ij}^{kk}\right)|i\rangle\langle j| \,. \tag{1.54}$$

Similarly, the partial trace over A becomes

$$\rho_B = \mathrm{tr}_A(\rho_{AB}) = \sum_{ijk\ell} \gamma_{ij}^{k\ell} \mathrm{tr}(|i\rangle\langle j|) \otimes |k\rangle\langle \ell| = \sum_{k\ell}\left(\sum_j \gamma_{jj}^{k\ell}\right)|k\rangle\langle \ell| \,. \tag{1.55}$$

The states ρ_A, ρ_B are also referred to as *reduced states*.

∎ **Example 1.6.2** Let's consider again the example of the EPR pair

$$|\mathrm{EPR}\rangle_{AB} = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle),$$

with associated density matrix $\rho_{AB} = |\mathrm{EPR}\rangle\langle\mathrm{EPR}|_{AB}$. Using the definition we can compute

$$\begin{aligned} \mathrm{tr}_B(\rho_{AB}) = \frac{1}{2}\big(&|0\rangle\langle 0| \otimes \mathrm{tr}(|0\rangle\langle 0|) + |0\rangle\langle 1| \otimes \mathrm{tr}(|0\rangle\langle 1|) \\ &+ |1\rangle\langle 0| \otimes \mathrm{tr}(|1\rangle\langle 0|) + |1\rangle\langle 1| \otimes \mathrm{tr}(|1\rangle\langle 1|)\big) \,. \end{aligned} \tag{1.56}$$

Since the trace is cyclic, $\mathrm{tr}(|0\rangle\langle 1|) = \langle 1|0\rangle = 0$, similarly $\mathrm{tr}(|1\rangle\langle 0|) = 0$, but $\mathrm{tr}(|0\rangle\langle 0|) = \mathrm{tr}(|1\rangle\langle 1|) = 1$ and hence

$$\mathrm{tr}_B(\rho_{AB}) = \frac{1}{2}\left(|0\rangle\langle 0| + |1\rangle\langle 1|\right) = \frac{\mathbb{I}}{2} \,. \tag{1.57}$$

Convince yourself that when we take the partial trace operation over A, and hence look at the state of just Bob's qubit we have

$$\mathrm{tr}_A(\rho_{AB}) = \frac{\mathbb{I}}{2}\,. \tag{1.58}$$

■

Exercise 1.6.2 If $\rho_{AB} = |\Phi\rangle\langle\Phi|$ is the singlet $|\Phi\rangle = (|01\rangle - |10\rangle)/\sqrt{2}$, compute ρ_A and ρ_B. ■

■ **Example 1.6.3** We can now see that performing a unitary operation on A has no effect on the state of B, i.e., it does not change ρ_B.

$$(U_A \otimes \mathbb{I}_B)\rho_{AB}(U_A \otimes \mathbb{I}_B)^\dagger = \sum_{ijk\ell} \gamma_{ij}^{k\ell} U_A|i\rangle\langle j|_A U_A^\dagger \otimes |k\rangle\langle\ell|_B. \tag{1.59}$$

Computing again the partial trace we have

$$\begin{aligned}
\mathrm{tr}_A(U_A \otimes \mathbb{I}_B \rho_{AB} U_A^\dagger \otimes \mathbb{I}_B) &= \sum_{ijk\ell} \gamma_{ij}^{k\ell}\, \mathrm{tr}(U_A|i\rangle\langle j|U_A^\dagger) \otimes |k\rangle\langle\ell| && (1.60)\\
&= \sum_{ijk\ell} \gamma_{ij}^{k\ell}\, \mathrm{tr}(|i\rangle\langle j|U_A^\dagger U_A) \otimes |k\rangle\langle\ell| && (1.61)\\
&= \sum_{ijk\ell} \gamma_{ij}^{k\ell}\, \mathrm{tr}(|i\rangle\langle j|) \otimes |k\rangle\langle\ell| && (1.62)\\
&= \sum_{k\ell} \left(\sum_j \gamma_{jj}^{k\ell} \right) |k\rangle\langle\ell| = \rho_B\,. && (1.63)
\end{aligned}$$

Can you convince yourself that performing a measurement on A also has no effect on B? ■

1.7 Secure message transmission

The first cryptographic challenge that we will consider is the one of secure message transmission. Here, our protagonists Alice and Bob want to protect their communication from the prying eyes of an eavesdropper Eve. Alice and Bob are always honest, and Eve is the *adversary* (sometimes also called *eavesdropper*. Alice and Bob have control over their secure labs that Eve cannot peek into. However, Eve has access to the communication channel connecting Alice and Bob.

The most fundamental (and also the most secure) method that Alice and Bob can use to transmit their messages securely requires them to use a *key* to encode the message. It is assumed that the key is known to both Alice and Bob, but is private to them: Eve has no information about the key. For this reason we call cryptosystems such as the one we're about to discover *private-key* cryptosystems. Today we investigate how such secret key can be used. In later weeks we will use quantum information to come up with the key!

1.7.1 Shannon's secrecy condition and the need for large keys

Let us assume that Alice and Bob share a classical key k that is unknown to the eavesdropper, in the sense that we will make precise later. For the moment, let us take the intuitive definition that Eve doesn't know the key if she is completely uncorrelated from the key, and $p(k) = 1/|K|$ for $|K|$ possible keys, i.e. every key is equally likely. A mathematical framework for the description of transferring secret messages was first developed in [Sha49]. Any encryption scheme consists of some encryption function $\mathrm{Enc}(k,m) = e$ that takes the key k and the message m and maps it to some encrypted message e. The original message m is also called the plaintext, and e the ciphertext. We will also need a decryption function $\mathrm{Dec}(k,e) = m$ that takes the key k and the cipertext e back to the plaintext.

Definition 1.7.1 An encryption scheme $(\mathrm{Enc}, \mathrm{Dec})$ is *secret*, or *secure* if and only if for all prior distributions $p(m)$ over messages, and all messages m, we have

$$p(m) = p(m|e), \tag{1.64}$$

where $e = \mathrm{Enc}(k, m)$.

In other words we call an encryption scheme secret/secure whenever an eavesdropper Eve who may have intercepted the ciphertext e gains no additional knowledge about the message m than she would have without the ciphertext e. That is, the probability $p(m)$ of the message m is the same a priori (as anyone could guess) as it is from the point of view of Eve , who has obtained e. This is a very strong notion of security: absolutely no information is gained by having access to e!

Note that it would be easy to come up with an encryption scheme which is "just" secret: Alice simply sends a randomly chosen e to Bob. At this point, you are probably objecting since surely this would not be very useful! How could Bob hope to learn m, if e has nothing to do with m? The second condition that an encryption scheme has to satisfy is thus that it is *correct*.

Definition 1.7.2 An encryption scheme $(\mathrm{Enc}, \mathrm{Dec})$ is *correct* if and only if for all possible messages m, and all possible keys k, we have $m = \mathrm{Dec}(k, \mathrm{Enc}(k, m))$.

Again it would be easy to find an encryption scheme that is "just" correct: Alice simply sends $e = m$ to Bob. Again, you are possibly objecting, since Eve can now read all messages and this is precisely what we wanted to prevent!

The art of cryptography is to design protocols that are *both* correct *and* secure simultaneously. In almost all situations, it will be easy to be correct, and easy to be secure, but the real challenge arises when we want to combine both conditions.

The secret key we assumed Alice and Bob share will be the essential ingredient required to achieve an encryption scheme that is both correct and secret. Is a key really needed? As it turns out, not only it is needed but in fact we will need just as many keys as there are possible messages. A message is called possible if $p(m) > 0$. Let us establish this fact in a lemma, due to Shannon:

Lemma 1 An encryption scheme $(\mathrm{Enc}, \mathrm{Dec})$ can only be *secure* and *correct* if the number of possible keys $|K|$ is at least as large as the number of possible messages $|M|$, that is, $|K| \geq |M|$.

Proof. Suppose for contradiction that there exists a scheme using less keys, i.e., $|K| < |M|$. We will show that such a scheme cannot be secure. Consider an eavesdropper who has intercepted the ciphertext e. She could then compute

$$\mathscr{S} = \{\hat{m} \mid \exists k, \hat{m} = \mathrm{Dec}(k, e)\} \ , \tag{1.65}$$

that is, the set of all messages $\hat{m}$ for which there exists a key k that could have resulted in the observed ciphertext e. Note that the size $|\mathscr{S}|$ of this set is $|\mathscr{S}| \leq |K|$, since for each possible key k we get at most one message $\hat{m}$. Since $|K| < |M|$, we thus have $|\mathscr{S}| < |M|$. This means that there exists at least one message m such that $m \notin \mathscr{S}$, and hence $p(m|e) = 0$. There is no key which could give this message, so the eavesdropper learns that the message cannot have been m, but one of the other messages instead! Since a message is possible precisely when $p(m) > 0$, we thus have $0 = p(m|e) \neq p(m) > 0$, which violates the security condition. We conclude that the scheme can only be secure if $|K| \geq |M|$. ■

Can the bound given in the lemma be achieved: does there exist an encryption scheme that is both correct *and* secure, and which uses precisely the minimal number of keys $|K| = |M|$? The answer is yes! We shall explore that in the next section.

1.8 The (quantum) one-time pad

Let us consider possibly the simplest scheme to encrypt messages. It is known as the one-time pad, and offers excellent security — we will learn precisely why soon!

1.8.1 The classical one-time pad

Imagine that Alice (the sender) wants to send a secret message m to Bob (the receiver), where we will take $m \in \{0,1\}^n$ to be an n-bit string. Let us furthermore imagine that Alice and Bob already share a key $k \in \{0,1\}^n$ which is just as long as the message. Indeed, as we have seen earlier, having a key that is as long as the message is a requirement to ensure absolute security for arbitrary messages m!

Protocol 1 The classical *one-time* pad is an encryption scheme in which the encryption of a message $m \in \{0,1\}^n$ using the key $k \in \{0,1\}^n$ is given by

$$\mathrm{Enc}(k,m) = m \oplus k = (m_1 \oplus k_1, m_2 \oplus k_2, \ldots, m_n \oplus k_n) = (e_1, \ldots, e_n) = e \,, \tag{1.66}$$

where $m_j \oplus k_j = m_j + k_j \mod 2$ is the XOR, or addition modulo 2. The decryption is given by

$$\mathrm{Dec}(k,e) = e \oplus k = (e_1 \oplus k_1, e_2 \oplus k_2, \ldots, e_n \oplus k_n). \tag{1.67}$$

Note that since $m_j \oplus k_j \oplus k_j = m_j$ the receiver can recover the message, thus the scheme is correct. Is it secure?

To see that it satisfies Shannon's definition, consider any message m. For a uniformly random choice of key k, the associated ciphertext $e = \mathrm{Enc}(k,m)$ is uniformly distributed over all n-bit strings. We have

$$p(e|m) = p(m \oplus k|m) = p(k|m) = \frac{1}{2^n} \,, \tag{1.68}$$

for $k = e \oplus m$. Now note that this holds for all messages m, and hence

$$p(e) = \sum_m p(m)p(e|m) = \frac{1}{2^n} \,. \tag{1.69}$$

Applying Bayes rule we thus have

$$p(m|e) = \frac{p(m,e)}{p(e)} = \frac{p(e|m)p(m)}{p(e)} = p(m) \,, \tag{1.70}$$

independent of m, and since $p(m|e) = p(m)$ the scheme is perfectly secure. Note however that the argument crucially relies on the key being uniformly distributed and independent from the eavesdropper, a condition that has to be treated with care. In week 4 we will learn about a method called *privacy amplification* that can be used to "improve" the quality of a key about which the eavesdropper may have partial information. We will make this notion precise later in this lecture series!

We note that while the one-time pad is perfectly secure, it does not protect against the eavesdropper changing bits in the mesages. For example, Eve can flip bits - while this may not bother you very much when transmitting images, it surely will be an issue in your bank transactions. For this reason, one-time pads are supplemented by checksums or message authentication codes (MAC) which allow changes to be detected (and corrected). These are purely classical techniques, and hence we will not cover them here.

There is another way to look at the classical one time pad that brings it much closer to the quantum version we will consider next. Let us explain this by considering the encryption of a single-bit message $m \in \{0,1\}$. Recall that we could encode the message into a quantum state as $|m\rangle$, or as the density matrix $|m\rangle\langle m|$. When we apply the XOR operation the result is that the bit m is flipped whenever the key bit $k = 1$. That is, when $k = 1$ we transform the state to $X|m\rangle$, or, as a density matrix, $X|m\rangle\langle m|X$. If Alice and Bob choose a random key bit k, then from the point of view of the eavesdropper (who does not have access to k) the state of the message is represented by the density matrix

$$\rho = \frac{1}{2}\sum_{k\in\{0,1\}} X^k \rho X^k = \frac{1}{2}|m\rangle\langle m| + \frac{1}{2}X|m\rangle\langle m|X = \frac{\mathbb{I}}{2}\,. \tag{1.71}$$

Note that this density matrix ρ does *not* depend on m! That is, absolutely no information about m can be gained from the density matrix that represents the eavesdropper's view of the system, i.e. the message m and any information held by the eavesdropper is uncorrelated. This "uncorrelated-ness" is precisely the desired hallmark of an encryption scheme, and you will soon learn how to make this precise!

1.8.2 The quantum one-time pad

Let us consider the task of encrypting a qubit, instead of a classical bit [Amb+00; BR00]. In the videos, we saw a geometric argument for encrypting a qubit. Here, we will give a formal argument. Instead of one key bit, however, it turns out that we require two key bits k_1k_2 to encrypt a qubit. Indeed, it can be shown that two key bits are *necessary*. An intuition on why we need more than one key bit is that we wish to hide information in all possible bases the qubit could be in. In the classical case applying the bit flip operator X allowed us to encrypt any bit expressed in the standard basis. If we are allowed other bases, we could for example attempt to encrypt a bit expressed in the Hadamard basis, in which case $X|+\rangle\langle+|X = |+\rangle\langle+|$ and $X|-\rangle\langle-|X = |-\rangle\langle-|$. In other words, the qubit is unchanged by the "encryption" procedure, and the scheme is completely insecure.

The trick to a quantum one-time pad is then to apply a bit flip in both bases, standard and Hadamard. This can be achieved by applying $X^{k_1}Z^{k_2}$. When k_1k_2 is chosen uniformly at random, an arbitrary single-qubit ρ is encrypted to

$$\frac{1}{4}\sum_{k_1,k_2\in\{0,1\}} X^{k_1}Z^{k_2}\rho Z^{k_2}X^{k_1}\,. \tag{1.72}$$

To see why this works, let us recall the Bloch sphere representation of ρ and the fact that the Pauli matrices pairwise anti-commute. In particular, applying either $\mathbb{I}$, X, Z or XZ with equal probability to the Pauli matrix X gives

$$X + XXX + ZXZ + XZXZX = X + X - ZZX - XZZXX = X + X - X - X = 0\,, \tag{1.73}$$

where we use the fact that the Pauli matrices are observables (i.e. they are Hermitian and square to identity), and $\{X,Z\} = XZ + ZX = 0$. For some intuition, refer to Figure 1.1 for a visualization.

Exercise 1.8.1 Show that for all $M \in \{X,Z,Y\}$ we have

$$\frac{1}{4}\sum_{k_1,k_2} X^{k_1}Z^{k_2}MZ^{k_2}X^{k_1} = 0. \tag{1.74}$$

■

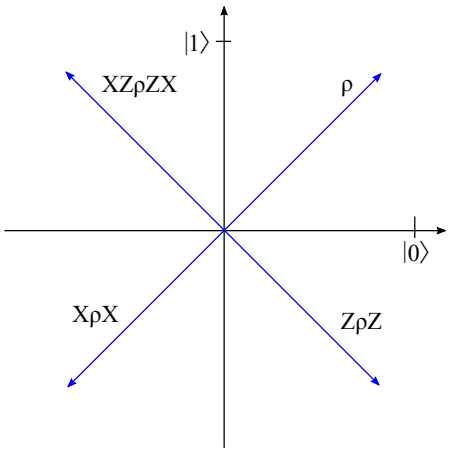

Figure 1.1: A qubit encoded by two key bits: the operations $\mathbb{I}, X, Z, XZ$ are performed on the qubit with equal probability. The resulting mixture of states is then the maximally mixed state (represented by the origin of the diagram).

Since we can write any one qubit state as

$$\rho = \frac{1}{2}\left(\mathbb{I} + v_x X + v_y Y + v_z Z\right) , \tag{1.75}$$

we thus have that

$$\frac{1}{4}\sum_{k_1,k_2} X^{k_1} Z^{k_2} \rho Z^{k_2} X^{k_1} = \frac{\mathbb{I}}{2} . \tag{1.76}$$

To someone who does not know k_1, k_2 the resulting state is again completely independent of the input ρ, which means we have managed to hide all possible information from the eavesdropper. We thus have the following encryption scheme.

Protocol 2 The quantum one-time pad is an encryption scheme for qubits. To encrypt, Alice applies $X^{k_1} Z^{k_2}$ to the qubit ρ and sends the resulting state to Bob. To decrypt, Bob applies the inverse $(X^{k_1} Z^{k_2})^\dagger$ to obtain ρ.

This scheme can be extended to n qubits, where on each qubit we apply either $\mathbb{I}$, X, Z or XZ depending on two key bits. This means that to encrypt n qubits, we use $2n$ bits of classical key.

Exercise 1.8.2 Show that strings of Pauli matrices $P^s = X^{s_1} Z^{s_2} \otimes X^{s_3} Z^{s_4} \otimes \ldots \otimes X^{s_{2n-1}} Z^{s_{2n}}$ with $s \in \{0,1\}^{2n}$ form an orthogonal basis for all linear operators $\mathscr{L}(\mathbb{C}^{2^n}, \mathbb{C}^{2^n})$, in which n-qubit density matrices ρ can be described. That is, $\mathrm{tr}[(P^s)^\dagger P^{\hat{s}}] = 0$ for all $s \neq \hat{s}$, and that we can write a density matrix on n qubits as

$$\rho = \frac{1}{2^n}\left(\mathbb{I}^{\otimes 2n} + \sum_{s \neq 0} v_s P^s\right) . \tag{1.77}$$

■

R It would be natural to think that for n-qubit systems as for 1-qubit systems the coefficients v_s associated with density matrices could be characterized by some form of higher-dimensional analogue of the Bloch sphere. This is not true, and much more complicated conditions on the coefficients v_s have to hold for ρ to be a valid quantum state. The Bloch sphere representation is only used for a single qubit, where it forms a useful visualization tool.

Acknowledgements

This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International Licence. The lecture notes are written by Stephanie Wehner, Nelly Ng, and Thomas Vidick. We thank David Elkouss, Jonas Helsen, Jérémy Ribeiro and Kenneth Goodenough for proofreading.

Important identities for calculations

Trace

Given a matrix M, the trace is given by $\mathrm{tr}(M) = \sum_i M_{ii}$, i.e. the sum of its diagonal elements. The trace operation is cyclic, i.e. for any two matrices M, N, $\mathrm{tr}(MN) = \mathrm{tr}(NM)$.

Density Matrices

If a source prepares a quantum system in the state ρ_x with probability p_x, then the resulting state of the system is given by the density matrix

$$\rho = \sum_x p_x \rho_x. \tag{1.78}$$

Bloch representation of density matrices: any qubit density matrix can be written as

$$\rho = \frac{1}{2}\left(\mathbb{I} + v_x X + v_z Z + v_y Y\right), \tag{1.79}$$

and the Bloch vector $\vec{v} = (x_x, v_y, v_z) \leq 1$ with equality if and only if ρ is pure.

Probability of measurement outcomes on a density matrix

If a quantum state with density matrix ρ is measured in the basis $\{|w_j\rangle\}_j$, then the probabilities of obtaining each outcome $|w_j\rangle$ is given by

$$p_{w_j} = \langle w_j|\rho|w_j\rangle = \mathrm{tr}(\rho|w_j\rangle\langle w_j|). \tag{1.80}$$

Combining density matrices

For density matrices $\rho_A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$ and $\rho_B = \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix}$ representing qubits A and B, the joint density matrix is given by

$$\rho_{AB} = \rho_A \otimes \rho_B := \begin{pmatrix} a_{11}\rho_B & a_{12}\rho_B \\ a_{21}\rho_B & a_{22}\rho_B \end{pmatrix} = \begin{pmatrix} a_{11}b_{11} & a_{11}b_{12} & a_{12}b_{11} & a_{12}b_{12} \\ a_{11}b_{21} & a_{11}b_{22} & a_{12}b_{21} & a_{12}b_{22} \\ a_{21}b_{11} & a_{21}b_{12} & a_{22}b_{11} & a_{22}b_{12} \\ a_{21}b_{21} & a_{21}b_{22} & a_{22}b_{21} & a_{22}b_{22} \end{pmatrix}. \tag{1.81}$$

Partial trace

Given a bipartite matrix ρ_{AB}, which can be expressed in a general form:

$$\rho_{AB} = \sum_{ijkl} \gamma_{ij}^{kl} |i\rangle\langle j| \otimes |k\rangle\langle l|, \tag{1.82}$$

the partial trace operation over system A yields the reduced state ρ_B

$$\rho_B = \mathrm{tr}_A(\rho_{AB}) = \sum_{ijk\ell} \gamma_{ij}^{k\ell}\, \mathrm{tr}(|i\rangle\langle j|) \otimes |k\rangle\langle \ell| = \sum_{k\ell} \left(\sum_j \gamma_{jj}^{k\ell}\right) |k\rangle\langle \ell|\,. \tag{1.83}$$

Properties of Pauli Matrices X, Z, Y

For any $S_1, S_2 \in \{X, Y, Z\}$, $\{S_1, S_2\} = 2\delta_{S_1 S_2}\mathbb{I}$ where the anti-commutator is $\{A, B\} = AB + BA$. This implies the following

1. Zero trace: $\mathrm{tr}(S_1) = 0$.
2. Orthogonality: $\mathrm{tr}(S_1^\dagger S_2) = 0$.
3. Unitary: $S_1^\dagger S_1 = S_1 S_1^\dagger = \mathbb{I}$.
4. Squared to identity: $S_1^2 = \mathbb{I}$.

Bibliography

[Amb+00] A. Ambainis et al. “Private quantum channels”. In: *Proceedings of FOCS*. arXiv:quant-ph/0003101. 2000 (cited on page 20).

[BR00] P. O. Boykin and V. Roychowdhury. “Optimal encryption of quantum bits”. quant-ph/0003059. 2000 (cited on page 20).

[Kel94] D.G. Kelly. *Introduction to Probability*. Macmillan Publishing Company, 1994. ISBN: 9780023631450 (cited on page 3).

[Ros10] S.M. Ross. *A First Course in Probability*. Pearson Prentice Hall, 2010. ISBN: 9780136033134 (cited on page 3).

[Sha49] Claude E Shannon. “Communication theory of secrecy systems”. In: *Bell system technical journal* 28.4 (1949), pages 656–715 (cited on page 17).

Lecture Notes

Quantum Cryptography Week 2: The Power of Entanglement

Contents

We already encountered quantum entanglement in the form of the EPR pair $|\text{EPR}\rangle = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$. This week we will define entanglement more formally and explore some of the reasons that make it such an interesting topic in quantum information. To wet your appetite, let it already be said that in later weeks we will see that entanglement allows us to guarantee the security of communications based only on the laws of nature. We also know that entanglement is a necessary ingredient in the most impressive quantum algorithms, such as Shor's algorithm for factoring, and for quantum error correction.

2.1 Entanglement

If we combine two qubits A and B, each of which is in a pure state, the joint state of the two qubits is given by

$$|\psi\rangle_{AB} = |\psi_1\rangle_A \otimes |\psi_2\rangle_B \ . \tag{2.1}$$

Any two-qubit state that is either directly of this form, or is a mixture of states of this form, is called *separable*. Entangled state are states which are *not* separable. In other words, a pure state $|\psi\rangle$ is entangled if and only if

$$|\psi\rangle \neq |\psi_1\rangle \otimes |\psi_2\rangle \ , \tag{2.2}$$

for any possible choice of $|\psi_1\rangle$ and $|\psi_2\rangle$. A mixed state ρ is entangled if and only if it cannot be written as a convex combination of pure product states of the form in Eq. (2.1).

■ **Example 2.1.1** An example of an entangled state of two qubits is the EPR pair

$$|\text{EPR}\rangle_{AB} = \frac{1}{\sqrt{2}}\left(|00\rangle_{AB} + |11\rangle_{AB}\right) \ . \tag{2.3}$$

When we learn more about entanglement later on, we will see that this state is, in a precise sense, the "most entangled" state of two qubits. The EPR pair is thus often referred to as a *maximally entangled* state. (There are other two-qubit states which are different from the EPR pair but have just about the same "amount" of entanglement; we will learn about these other maximally entangled states later.) ■

We have already seen that the EPR pair has the special property that it can be written in many symmetric ways. For instance, in the Hadamard basis

$$\frac{1}{\sqrt{2}}\left(|++\rangle_{AB} + |--\rangle_{AB}\right) \ . \tag{2.4}$$

Thus measurements of both qubits in the standard basis, or the Hadamard basis, always produce the same outcome. In a few weeks we will see that this property can even be used to *characterize* the EPR pair: it is the only two-qubit state having this property!

Exercise 2.1.1 Suppose that ρ_{AB} is a two-qubit separable state. Show that if a measurement of both qubits of ρ_{AB} in the standard basis always yields the same outcome, then a measurement of both qubits in the Hadamard basis necessarily has non-zero probability of giving different outcomes. Deduce a proof that the EPR pair (2.3) is not a separable state. ■

Entanglement has another interesting property which we will see later, called "monogamy". Monogamy states that if two systems are maximally entangled with each other then they cannot have any entanglement with any other system: equivalently, they must be in tensor product with the remainder of the universe.

■ **Example 2.1.2** Consider the state

$$|\psi\rangle_{AB} = \frac{1}{\sqrt{2}}(|01\rangle_{AB} + |11\rangle_{AB}) \ . \tag{2.5}$$

In contrast to the EPR pair in Example 2.1.1 this state is not entangled, since $|\psi\rangle_{AB} = \frac{1}{\sqrt{2}}(|0\rangle_A + |1\rangle_A) \otimes |1\rangle_B = |+\rangle_A \otimes |1\rangle_B$. ■

Definition 2.1.1 — Entanglement. Consider two quantum systems A and B. The joint state ρ_{AB} is *separable* if there exists a probability distribution $\{p_i\}_i$, and sets of density matrices $\{\rho_i^A\}_i, \{\rho_i^B\}_i$ such that

$$\rho_{AB} = \sum_i p_i \rho_i^A \otimes \rho_i^B. \tag{2.6}$$

If there exists no such decomposition ρ_{AB} is called *entangled*.
If $\rho_{AB} = |\Psi\rangle\langle\Psi|_{AB}$ is a pure state, then $|\Psi\rangle_{AB}$ is separable if and only if there exists $|\psi\rangle_A, |\psi\rangle_B$ such that

$$|\Psi\rangle_{AB} = |\psi\rangle_A \otimes |\psi\rangle_B. \tag{2.7}$$

■ **Example 2.1.3** Consider the density matrix

$$\rho_{AB} = \frac{1}{2}|0\rangle\langle 0|_A \otimes |1\rangle\langle 1|_B + \frac{1}{2}|+\rangle\langle +|_A \otimes |-\rangle\langle -|_B. \tag{2.8}$$

Such a state is in the form of Eq. (2.6), so it is not entangled: it is separable. Note that this does not imply that the systems A and B are necessarily independent: here they are correlated, but not entangled. (We would typically say that they are "classically correlated".) ■

■ **Example 2.1.4** Any cq-state, i.e. a state of the form $\rho_{XQ} = \sum_i p_i |x\rangle\langle x|_X \otimes \rho_x^Q$, is separable. ■

It is important to make the distinction between the two states

$$\rho_{AB} = \frac{1}{2}|0\rangle\langle 0|_A \otimes |0\rangle\langle 0|_B + \frac{1}{2}|1\rangle\langle 1|_A \otimes |1\rangle\langle 1|_B \qquad \text{and} \qquad \sigma_{AB} = |\text{EPR}\rangle\langle\text{EPR}|_{AB}. \tag{2.9}$$

For the state ρ_{AB}, if A is measured in the standard basis then whenever $|0\rangle_A$ is observed the state on B is $|0\rangle_B$; likewise when $|1\rangle_A$ is observed, the state on B is $|1\rangle_B$. This is also true for σ_{AB}. However, consider measuring system A of ρ_{AB} in the Hadamard basis. The corresponding measurement operators are $|+\rangle\langle +|_A \otimes \mathbb{I}_B, |-\rangle\langle -|_A \otimes \mathbb{I}_B$. The post-measurement state conditioned on obtaining the outcome $|+\rangle_A$ is then

$$\rho_{|+_A}^{AB} = \frac{(|+\rangle\langle +|_A \otimes \mathbb{I}_B)\rho_{AB}(|+\rangle\langle +|_A \otimes \mathbb{I}_B)}{\text{tr}((|+\rangle\langle +|_A \otimes \mathbb{I}_B)\rho_{AB})} \tag{2.10}$$

$$= 2 \cdot \left(\frac{1}{2}\frac{1}{2}|+\rangle\langle +|_A \otimes |0\rangle\langle 0|_B + \frac{1}{2}\frac{1}{2}|+\rangle\langle +|_A \otimes |1\rangle\langle 1|_B\right) \tag{2.11}$$

$$= |+\rangle\langle +|_A \otimes \frac{\mathbb{I}_B}{2}, \tag{2.12}$$

and we see that the reduced state on B, $\rho_{|+_A}^B = \frac{\mathbb{I}_B}{2}$ is maximally mixed. In contrast, using that the state $|\text{EPR}\rangle_{AB}$ can be rewritten as

$$|\text{EPR}\rangle_{AB} = \frac{1}{\sqrt{2}}(|00\rangle_{AB} + |11\rangle_{AB}) = \frac{1}{\sqrt{2}}(|++\rangle_{AB} + |--\rangle_{AB}), \tag{2.13}$$

when σ_{AB} is measured with respect to the Hadamard basis on system A, conditioned on the outcome $|+\rangle_A$, the reduced state on B is $\sigma_{|+_A}^B = |+\rangle\langle +|_B$. In particular this state is pure: it is very different from the totally mixed state we obtained by performing the same experiment on ρ_{AB}. This is a sense in which the correlations in σ_{AB} are stronger than those in ρ_{AB}.

2.2 Purifications

Last week we learned about the partial trace operation, which provides a way to describe the state of a subsystem when given a description of the state on a larger composite system. Even if the state of the larger system is pure, the reduced state can sometimes be mixed, and this is a signature of entanglement in the larger state.

Is it possible to reverse this process? Suppose given a density matrix ρ_A describing a quantum state on system A. Is it always possible to find a pure state $\rho_{AB} = |\Psi\rangle\langle\Psi|_{AB}$ such that $\mathrm{tr}_B(\rho_{AB}) = \rho_A$? Such a state is called a *purification* of ρ_A.

Definition 2.2.1 — Purification. Given any density matrix ρ_A, a pure state $|\Psi_{AB}\rangle$ is a *purification* of A if $\mathrm{tr}_B(|\Psi\rangle\langle\Psi|_{AB}) = \rho_A$.

Let's see how an arbitrary density matrix ρ_A can be purified. As a first step, diagonalize ρ_A, expressing it as a mixture

$$\rho_A = \sum_{j=1}^{d_A} \lambda_j^2 |\phi_j\rangle\langle\phi_j| \,, \tag{2.14}$$

where λ_j^2 are the (necessarily non-negative) eigenvalues of ρ_A and $|\phi_j\rangle$ the eigenstates. Since ρ_A is a density matrix the λ_j are non-negative and sum to 1. We've seen an interpretation of density matrices before: here we would say that ρ_A describes a quantum system that is in a probabilistic mixture of being in state $|\phi_j\rangle$ with probability λ_j^2. But who "controls" which part of the mixture A is in?

Let's introduce an imaginary system B which achieves just this. Let $\{|j\rangle_B\}_{j\in\{1,\ldots,d_B\}}$ be the standard basis for a system B of dimension $d_B = d_A$, and consider the pure state

$$|\Psi\rangle_{AB} = \sum_{j=1}^{d_A} \lambda_j |\phi_j\rangle_A \otimes |j\rangle_B \,, \tag{2.15}$$

where $\{|j\rangle_B\}_j$ is the standard basis on system B. Suppose we were to measure the B system of $|\Psi\rangle_{AB}$ in the standard basis. We know what would happen: we will obtain outcome j with probability $\langle\Psi|_{AB} M_j |\Psi\rangle_{AB}$, where $M_j = \mathbb{I}_A \otimes |j\rangle\langle j|_B$, and a short calculation will convince you this equals λ_j^2. Since we're using a projective measurement, we can describe the post-measurement state easily as being proportional to $M_j|\Psi\rangle\langle\Psi|_{AB}M_j$, and looking at the A system only we find that it is $|\phi_j\rangle\langle\phi_j|_A$.

To summarize, a measurement of system B gives outcome j with probability λ_J, and the post-measurement state on A is precisely $|\phi_j\rangle\langle\phi_j|$. This implies that $\mathrm{Tr}_B(|\Psi\rangle\langle\Psi|_{AB}) = \rho_A$, a fact which can be verified directly using the mathematical definition of the partial trace operation.

Are purifications unique? You'll notice that in the above construction we made the choice of the standard basis for system B, but any other basis would have worked just as well. So it seems like we at least have a choice of basis on system B: there is a "unitary degree of freedom". To see that this is the only freedom that we have in choosing a purification, we first need to learn about a very convenient representation of bipartite pure states, the *Schmidt decomposition*.

2.2.1 The Schmidt decomposition

The purification that we constructed in (2.15) has a special form: it is expressed as a sum, with non-negative coefficients whose squares sum to 1, of tensor products of basis states for the A and B systems respectively. As we saw, this particular form is convenient because it lets us compute the reduced states in A and B very easily. Unfortunately, not every state is always given in this way: for example, if we write $|\Psi\rangle_{AB} = \frac{1}{\sqrt{2}}(|0\rangle_A|0\rangle_B + |+\rangle_A|1\rangle_B)$ then the two states $|0\rangle_A, |+\rangle_A$ on A are not orthogonal. But maybe the same state can be written in a more convenient form? The answer is yes, and it is given by the Schmidt decomposition.

Theorem 2.2.1 — Schmidt decomposition. Consider quantum systems A and B with dimensions d_A, d_B respectively, and let $d = \min(d_A, d_B)$. Any pure bipartite state $|\Psi\rangle_{AB}$ has a Schmidt decomposition

$$|\Psi\rangle_{AB} = \sum_{i=1}^{d} \lambda_i |u_i\rangle_A |v_i\rangle_B, \tag{2.16}$$

where $\lambda_i \geq 0$ and $\{|u_i\rangle_A\}_i, \{|v_i\rangle_B\}_i$ are orthonormal vector sets. The coefficients λ_i are called the *Schmidt coefficients* and $|u_i\rangle_A, |v_i\rangle_B$ the *Schmidt vectors*.

We discussed the proof of the theorem in the video module; you can also find a detailed proof in Section 2.5 of [NC01]. The main idea is to start by expressing $|\Psi\rangle_{AB} = \sum_{j,k} \alpha_{j,k} |j\rangle_A |k\rangle_B$ using the standard bases of A and B, and then write the singular value decomposition of the $d_A \times d_B$ matrix with coefficients $\alpha_{j,k}$ to recover the λ_i (the singular values) and the $|u_i\rangle_A$ (the left eigenvectors) and the $|v_i\rangle_B$ (the right eigenvectors).

The Schmidt decomposition has many interesting consequences. A first consequence is that it provides a simple recipe for computing the reduced density matrices: given a state of the form (2.16), we immediately get $\rho_A = \sum_i \lambda_i^2 |u_i\rangle\langle u_i|_A$, and $\rho_B = \sum_i \lambda_i^2 |v_i\rangle\langle v_i|_B$. An important observation is that ρ_A and ρ_B have the same eigenvalues, which are precisely the squares of the Schmidt coefficients. As a consequence, given any two density matrices ρ_A and ρ_B, there exists a pure bipartite state $|\Psi\rangle_{AB}$ such that $\rho_A = \mathrm{Tr}_B(|\Psi\rangle\langle\Psi|_{AB})$ and $\rho_B = \mathrm{Tr}_A(|\Psi\rangle\langle\Psi|_{AB})$ if and only if ρ_A and ρ_B have the same spectrum! Without the Schmidt decomposition this is not at all an obvious fact to prove.

The same observation also implies that the Schmidt coefficients are uniquely defined: they are the square roots of the eigenvalues of the reduced density matrix. The Schmidt vectors are also unique, up to degeneracy and choice of phase: if an eigenvalue has an associated eigenspace of dimension 1 only then the associated Schmidt vector must be the corresponding eigenvector. If the eigenspace has dimension more than 1 we can choose as Schmidt vectors any basis for the subspace. And note that in (2.16) we can always multiply $|u_i\rangle$ by $e^{i\theta_i}$, and $|v_i\rangle$ by $e^{-i\theta_i}$, so there is a phase degree of freedom.

Another important consequence of the Schmidt decomposition is that it provides us with a way to measure entanglement between the A and B systems in a pure state $|\Psi_{AB}\rangle$. A first, rather rough but convenient such measure is given by the number of non-zero coefficients λ_j. This measure is the so-called *Schmidt rank*. If the Schmidt rank is 1 then the state is a product state, and if it is strictly larger than 1 then the state is entangled.

Definition 2.2.2 — Schmidt rank. For any bipartite pure state with Schmidt decomposition $|\Psi\rangle_{AB} = \sum_{i=1}^{d} \lambda_i |a_i\rangle_A |b_i\rangle_B$, the *Schmidt rank* is defined as the number of non-zero coefficients λ_i. It is also equal to $\mathrm{rank}(\rho_A)$ and $\mathrm{rank}(\rho_B)$.

The Schmidt coefficients provide a finer way to measure entanglement than the Schmidt rank. A natural measure, called "entropy of entanglement", consists in taking the entropy of the distribution specified by the squares of the coefficients. If the entropy is 0 then there is only a single coefficient equal to 1, and the state is not entangled. But as soon as the entropy is positive the state is entangled. This measure is finer than the Schmidt rank. For example, it distinguishes the entanglement in the two states

$$|\Psi\rangle = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle \qquad \text{and} \qquad |\phi\rangle = \sqrt{1-\varepsilon}|00\rangle + \sqrt{\varepsilon}|11\rangle.$$

For small $0 < \varepsilon < 1/2$ both states have the same Schmidt rank, but the first one has entanglement entropy 1 whereas the second has entanglement entropy $H(\varepsilon)$ (where H is the binary entropy function) going to 0 as $\varepsilon \to 0$. This is the reason why we call the EPR pair "maximally entangled":

its entanglement entropy is maximal among all two-qubit states.

2.2.2 Uhlmann's theorem

Let's return to the topic of the freedom in choosing purifications of a density matrix. We saw that we at least had a unitary degree of freedom by choosing a basis on the purifying system B. Uhlmann's theorem states that this is precisely the only freedom we have.

Theorem 2.2.2 — Uhlmann's theorem. Suppose given a density matrix ρ_A and a purification of A given by $|\Psi\rangle_{AB}$. Then another state $|\Phi\rangle_{AB}$ is also a purification of A if and only if there exists a unitary U_B such that

$$|\Phi\rangle_{AB} = \mathbb{I}_A \otimes U_B |\psi\rangle_{AB}. \tag{2.17}$$

We already saw a proof of the "if" part of the theorem. To show the converse, i.e. that two purifications must always be related by a unitary, consider the Schmidt decomposition:

$$|\Phi\rangle_{AB} = \sum_i \lambda_i |u_i\rangle_A |v_i\rangle_B,$$
$$|\Psi\rangle_{AB} = \sum_i \mu_i |w_i\rangle_A |z_i\rangle_B.$$

As we know the λ_i are uniquely defined: they are the square roots of the eigenvalues of ρ_A. So if $|\Phi\rangle_{AB}$ and $|\Psi\rangle_{AB}$ are both purifications of the same ρ_A, we must have $\lambda_i = \mu_i$. Now suppose for simplicity that all eigenvalues are non-degenerate. Then the $|u_i\rangle_A$ are also uniquely determined: they are the eigenvectors of ρ_A associated to the λ_i. Therefore $|u_i\rangle_A = |w_i\rangle_A$ as well! Thus we see that the only choice we have left are the $|v_i\rangle_B$, or $|z_i\rangle_B$: since the density matrix ρ_B of the purification is not specified a priori, we may choose any orthonormal basis of the B system. Since any two orthonormal bases of the same space are related by a unitary matrix, this choice of basis is precisely the degree of freedom that is guaranteed by Uhlmann's theorem.

2.3 Secret sharing

Let's discuss a cryptographic application of the notions we just introduced. The application is called *secret sharing*. Imagine a country owns nuclear weapons yet wants to make sure that both the queen (Alice) and king (Bob) have to come together to activate them. One solution would be to give half of the launch codes $s = (s_1, \ldots, s_\ell) \in \{0,1\}^\ell$ to Alice, and the other half to Bob, thereby making sure that they both need to reveal their share of the information in order for the weapons to be activated. A drawback of this scheme is that each of them does have significant information about the launch codes, namely half of the bits. And what if there is only one bit? (Although that wouldn't be very secure, would it...)

The goal in a secret sharing scheme is to divide the information s into shares in such a way that any unauthorized set of parties (in the example, Alice or Bob alone) cannot learn *anything* at all about the secret. Remembering the idea behind the one-time pad, a much better scheme would be to choose a random string $r \in \{0,1\}^\ell$ and give r to Alice and $r \oplus s$ to Bob. In this case neither Alice nor Bob individually has any information about s; their respective secrets appear uniformly random. Yet when they come together they can easily recover s!

From the example above we see that given a random classical bit one can construct a secret sharing scheme between Alice and Bob that shares a single secret bit s. However they can do better if they are each given a qubit instead. Consider the case that Alice and Bob are given one of the

following four states at random:

$$|\psi_{00}\rangle_{AB} = \frac{1}{\sqrt{2}}(|00\rangle_{AB} + |11\rangle_{AB}), \qquad |\psi_{01}\rangle_{AB} = \frac{1}{\sqrt{2}}(|00\rangle_{AB} - |11\rangle_{AB}), \tag{2.18}$$

$$|\psi_{10}\rangle_{AB} = \frac{1}{\sqrt{2}}(|01\rangle_{AB} + |10\rangle_{AB}), \qquad |\psi_{11}\rangle_{AB} = \frac{1}{\sqrt{2}}(|01\rangle_{AB} - |10\rangle_{AB}). \tag{2.19}$$

These states are called the *Bell states*. Observe that they are orthonormal and thus form a basis of $\mathbb{C}^2 \otimes \mathbb{C}^2$. We've already calculated the reduced density on Alice's system of one of those states, the EPR pair $|\psi_{00}\rangle_{AB}$:

$$\begin{aligned}
\rho_{00}^A &= \mathrm{tr}_B(|\psi_{00}\rangle\langle\psi_{00}|_{AB}) \\
&= \frac{1}{2}\big(|0\rangle\langle 0|_A \,\mathrm{tr}_B(|0\rangle\langle 0|_B) + |0\rangle\langle 1|_A \,\mathrm{tr}_B(|0\rangle\langle 1|_B) \\
&\qquad + |1\rangle\langle 0|_A \,\mathrm{tr}_B(|1\rangle\langle 0|_B) + |1\rangle\langle 1|_A \,\mathrm{tr}_B(|1\rangle\langle 1|_B)\big) \\
&= \frac{1}{2}(|0\rangle\langle 0|_A + |1\rangle\langle 1|_A) = \frac{\mathbb{I}_A}{2}.
\end{aligned}$$

Calculating the reduced states on either A or B for each each of these states always gives the same result,

$$\rho_{00}^A = \rho_{01}^A = \rho_{10}^A = \rho_{11}^A = \frac{\mathbb{I}}{2}, \tag{2.20}$$

$$\rho_{00}^B = \rho_{01}^B = \rho_{10}^B = \rho_{11}^B = \frac{\mathbb{I}}{2}. \tag{2.21}$$

We know what this means: since the reduced state on each subsystem is maximally mixed, neither Alice nor Bob can gain any information on which of the states $|\psi_{00}\rangle_{AB}, |\psi_{01}\rangle_{AB}, |\psi_{10}\rangle_{AB}, |\psi_{11}\rangle_{AB}$ they have one qubit of! However, due to the fact that these states together form a basis, when Alice and Bob come together they can perform a measurement in that basis that perfectly distinguishes which state they have, yielding two bits of information.

Exercise 2.3.1 Suppose there are now three parties, Alice, Bob and Charlie (the prime minister is also given a share of the nuclear codes!). Give a secret sharing scheme, based on a tripartite entangled state, such that no individual party has any information about the secret but the three of them together are able to recover the secret. Better: can you give a scheme such that no two of them has any information about the secret. Different: give a scheme such that no individual has any information about the secret, but any group of two can recover it. ∎

Application: Superdense coding

A different application of the usefulness of entanglement is to *superdense coding*. The task in dense coding consists in sending classical bits of information from Alice to Bob by encoding them in a quantum state that is as small as possible. Let's see how using entanglement we can send two classical bits using a single qubit.

Suppose Alice and Bob share the state $|\psi_{00}\rangle_{AB} = \frac{1}{\sqrt{2}}(|00\rangle_{AB} + |11\rangle_{AB})$, and that Alice performs a unitary on her qubit as indicated in Table 2.1, depending on which bits $ab \in \{00, 01, 10, 11\}$ she wants to send to Bob.

As we already saw, the four states on the right-hand side in Table 2.1 form the Bell basis, and in particular they are perfectly distinguishable. Hence if Alice sends her qubit over to Bob, he can perform a measurement in the Bell basis and recover both of Alice's classical bits.

Classical information a,b	Unitary $X_A^a Z_A^b$	Final joint state
00	$\mathbb{I}_A$	$\frac{1}{\sqrt{2}}(\|00\rangle_{AB} + \|11\rangle_{AB})$
01	X_A	$\frac{1}{\sqrt{2}}(\|10\rangle_{AB} + \|01\rangle_{AB})$
10	Z_A	$\frac{1}{\sqrt{2}}(\|00\rangle_{AB} - \|11\rangle_{AB})$
11	$-X_A Z_A$	$\frac{1}{\sqrt{2}}(\|01\rangle_{AB} - \|10\rangle_{AB})$

Table 2.1: Unitary operation performed by Alice in order to encode her two classical bits $ab \in \{0,1\}^2$.

2.4 Bell-Nonlocality

Entanglement has many counter-intuitive properties, many of which we will discover during the course of this lecture series. A very important one is that it allows correlations between two particles — two qubits — that cannot be replicated classically. The very first example of such correlations was demonstrated in [Bel64], where Bell proved that the predictions of quantum theory are incompatible with those of any classical theory satisfying a natural notion of *locality*.

The modern way to understand Bell non-locality is by means of so-called non-local games (see [Bru+14] for a detailed review on Bell non-locality). Let's imagine that we play a game with two players, which we'll again call Alice and Bob. Alice has a system A, and Bob has some system B. In this game, we will ask Alice and Bob questions, and collect answers. Let us denote the possible questions to Alice and Bob x and y, and label the answers a and b. We will play this game many times, and in each round choose the questions to ask with some probability $p(xy)$. As you might have guessed our little game has some rules. We denote these rules using a predicate $V(a,b|x,y)$, which takes the value "1" if a and b are winning answers for questions x and y. To be fair, Alice and Bob know the rules of the game given by $V(a,b|x,y)$, and also the distribution $p(xy)$. They can agree on any strategy before the game starts. However, once we start asking questions they are no longer allowed to communicate. Of interest to us will be the probability that Alice and Bob win the game, maximized over all possible strategies. That is,

$$p_{\text{win}} = \max_{\text{strategy}} \sum_{x,y} p(x,y) \sum_{a,b} V(a,b|x,y)\, p(a,b|x,y) \ , \tag{2.22}$$

where $p(a,b|x,y)$ is the probability that Alice and Bob produce answers a and b given x and y according to their chosen strategy.

What are these strategies? In a classical world, Alice and Bob can only have a classical strategy. A deterministic classical strategy is simply given by functions $f_A(x) = a$ and $f_B(y) = b$ that take the questions x and y to answers a and b. We then have $p(a,b|x,y) = 1$ whenever $a = f_A(x)$ and $b = f_B(y)$, and $p(a,b|x,y) = 0$ otherwise. Possibly, Alice and Bob also use shared randomness. That is, they have another string r, which they share with probability $p(r)$. In physics, r is also referred to as a hidden variable, but we will take the more operational viewpoint of shared randomness. In a strategy using shared randomness r, classical Alice and Bob can however still only apply functions: $a = f_A(x,r)$ and $b = f_B(y,r)$. In terms of the probabilities we then have $p(a,b|x,y,r) = 1$ if $a = f_A(x,r)$ and $b = f_B(y,r)$ and $p(a,b|x,y,r) = 0$ otherwise. This gives

$$p(a,b|x,y) = \sum_r p(r) p(a,b|x,y,r) \ . \tag{2.23}$$

Does shared randomness help Alice and Bob? Note that for a classical strategy based on shared

randomness we have

$$p_{\text{win}} = \max_{class.strat.} \sum_{x,y} p(x,y) \sum_{a,b} V(a,b|x,y) \sum_{r} p(r) p(a,b|x,y,r) \tag{2.24}$$

$$= \max_{class.strat.} \sum_{r} p(r) \Big(\sum_{x,y} p(x,y) \sum_{a,b} V(a,b|x,y)\, p(a,b|x,y,r) \Big) \,. \tag{2.25}$$

Note that the quantity in brackets is largest for some particular value(s) of r. Since Alice and Bob want to maximize their winning probability, they can thus fix the best possible r giving a deterministic strategy $a = f_A(x,r)$ and $b = f_A(y,r)$ where r is now fixed.

Why would we care about this at all? It turns out that for many games, a *quantum* strategy can achieve a higher winning probability. This is of fundamental importance for our understanding of nature. What's more, however, observing a higher winning probability is a signature of entanglement: quantumly, Alice and Bob can achieve a higher winning probability *only* if they are entangled, making such games into *tests* for entanglement. Testing whether the stated shared by Alice and Bob is entangled forms a crucial element in quantum key distribution, as we will see in later weeks.

Specifically, a *quantum strategy* means that Alice and Bob can pick a state ρ_{AB} to share, and agree on measurements to perform depending on their respective questions. That is, x and y will label a choice of measurement, and a and b are the outcomes of that measurement.

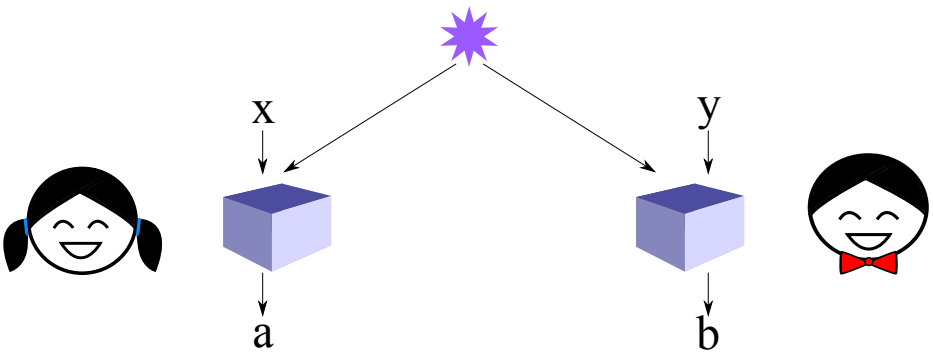

Figure 2.1: A non-local game. Alice and Bob are given questions x and y, and must return answers a and b. If Alice and Bob are quantum, then x and y label measurement settings and a and b are measurement outcomes.

2.4.1 Example of a non-local game: CHSH

Let us have a look at a very simple game based on the famous CHSH inequality. It will turn out to be extremely useful for quantum cryptography. At the start of the game, we send two bits x and y to Alice and Bob respectively, where we choose x with uniform probabilities $p(x=0) = p(x=1) = 1/2$ and y with probabilities $p(y=0) = p(y=1) = 1/2$. In turn, Alice and Bob will return answer bits a and b. Alice and Bob win the game if and only if

$$x \cdot y = a + b \mod 2 \,. \tag{2.26}$$

In terms of the predicate $V(a,b|x,y)$ this means that $V(a,b|x,y) = 1$ if $x \cdot y = a + b \mod 2$ and $V(a,b|x,y) = 0$ otherwise. We are interested in the probability that Alice and Bob win the game. This probability can be written as

$$p_{\text{win}}^{\text{CHSH}} = \frac{1}{4} \sum_{x,y \in \{0,1\}} \sum_{\substack{a,b \\ a+b \mod 2 = x \cdot y}} p(a,b|x,y) \,, \tag{2.27}$$

where $p(a,b|x,y)$ is the probability that Alice and Bob answer a and b given questions x and y. What can Alice and Bob do to win this game?

Classical winning probability

Classically, a is simply a function of x. For example, if $x=0$, then Alice and Bob could agree as part of their strategy that Alice will then always answer $a=0$. We see that as long as $x=0$ or $y=0$, then $x\cdot y=0$. In this case, Alice and Bob want to achieve $a+b \mod 2=0$. However, if $x=y=1$ then they would like to give answers such that $a+b \mod 2=1$. What makes this difficult for Alice and Bob is that they cannot communicate during the game. This means in particular that Alice's answer a can only depend on x (but not on y) and similarly Bob's answer b can only depend on y (but not on x).

It it not difficult to see (you may wish to check!) by trying out all possible strategies for Alice and Bob, that classically the maximum winning probability that can be achieved is

$$p_{\text{win}}^{\text{CHSH}}=\frac{3}{4}\,. \tag{2.28}$$

Alice and Bob can achieve this winning probability with the strategy of answering $a=b=0$ always, which means $a+b \mod 2=0$, which is correct in 3 out of the 4 possible cases. Only when $x=y=1$ will Alice and Bob make a mistake.

Quantum winning probability

It turns out that Alice and Bob can do significantly better with a quantum strategy, using shared entanglement. Indeed, suppose that Alice and Bob share an EPR pair, where we label the qubit held by Alice (A) and the one held by Bob (B).

$$|\Psi\rangle_{AB}=\frac{1}{\sqrt{2}}\left(|0\rangle_A|0\rangle_B+|1\rangle_A|1\rangle_B\right). \tag{2.29}$$

Suppose now that when $x=0$, Alice measures her qubit in the basis $\{|0\rangle,|1\rangle\}$. Otherwise when $x=1$, she measures in the basis $\{|+\rangle,|-\rangle\}$. Suppose furthermore that when $y=0$, Bob measures his qubit in the basis $|v_1\rangle,|v_2\rangle$ where

$$|v_1\rangle=\cos(\pi/8)|0\rangle+\sin(\pi/8)|1\rangle,\qquad |v_2\rangle=-\sin(\pi/8)|0\rangle+\cos(\pi/8)|1\rangle, \tag{2.30}$$

and when $y=1$, he measures in the basis $|w_1\rangle,|w_2\rangle$, where

$$|w_1\rangle=\cos(\pi/8)|0\rangle-\sin(\pi/8)|1\rangle,\qquad |w_2\rangle=\sin(\pi/8)|0\rangle+\cos(\pi/8)|1\rangle. \tag{2.31}$$

Consider the case where $x=0,y=0$. This means Alice measures in the basis $\{|0\rangle,|1\rangle\}$ and Bob in the basis $\{|v_1\rangle,|v_2\rangle\}$. The probability of winning, conditioned on $x=0,y=0$ is given by

$$\begin{aligned} p_{\text{win}|x=0,y=0} &= p(a=0,b=0|x=0,y=0)+p(a=1,b=1|x=0,y=0) && (2.32)\\ &= |\langle 0_A v_{1B}|\Psi_{AB}\rangle|^2+|\langle 1_A v_{2B}|\Psi_{AB}\rangle|^2 && (2.33)\\ &= 2\left|\frac{1}{\sqrt{2}}\cos\frac{\pi}{8}\right|^2=\cos^2\frac{\pi}{8}. && (2.34)\end{aligned}$$

The probability of winning, conditioned on $x=0,y=1$ is given by a similar expression

$$\begin{aligned} p_{\text{win}|x=1,y=0} &= p(a=0,b=0|x=1,y=0)+p(a=1,b=1|x=1,y=0) && (2.35)\\ &= |\langle 0_A w_{1B}|\Psi_{AB}\rangle|^2+|\langle 1_A w_{2B}|\Psi_{AB}\rangle|^2 && (2.36)\\ &= 2\left|\frac{1}{\sqrt{2}}\cos\frac{\pi}{8}\right|^2=\cos^2\frac{\pi}{8}. && (2.37)\end{aligned}$$

On the other hand,

$$p_{\text{win}|x=0,y=1} = p(a=0,b=0|x=0,y=1) + p(a=1,b=1|x=0,y=1) \tag{2.38}$$
$$= |\langle +_A v_{1B}|\Psi_{AB}\rangle|^2 + |\langle -_A v_{2B}|\Psi_{AB}\rangle|^2 \tag{2.39}$$
$$= \frac{1}{2}\left(\frac{1}{\sqrt{2}}\cos\frac{\pi}{8} + \frac{1}{\sqrt{2}}\sin\frac{\pi}{8}\right)^2 + \frac{1}{2}\left(\frac{1}{\sqrt{2}}\cos\frac{\pi}{8} + \frac{1}{\sqrt{2}}\sin\frac{\pi}{8}\right)^2 \tag{2.40}$$
$$= \frac{1}{2}\left(\cos\frac{\pi}{8} + \sin\frac{\pi}{8}\right)^2. \tag{2.41}$$

Finally, you may easily verify that for $x = y = 1$, $p_{\text{win}|x=1,y=1} = p_{\text{win}|x=1,y=0} = \frac{1}{2}\left(\cos\frac{\pi}{8} + \sin\frac{\pi}{8}\right)^2$. Also, convince yourself that $\frac{1}{2}\left(\cos\frac{\pi}{8} + \sin\frac{\pi}{8}\right)^2 = \cos^2\frac{\pi}{8}$. This implies that

$$p_{\text{win}} = \frac{1}{4}\sum_{x,y} p_{\text{win}|x,y} = \cos^2\frac{\pi}{8} \approx 0.85. \tag{2.42}$$

2.4.2 Implications

This counterintuitive effect of entanglement has far reaching consequences. The first is of a rather conceptual nature, as you may have started wondering what actually happens if we "measure" a quantum particle. Could it be that every particle has a local classical "cheat sheet" attached to it, which specifies the outcome it will give for any possible measurement that we can make on it? Such a cheat sheet would correspond precisely to a classical strategy in the game above: For every x, Alice's qubit has some outcome a attached. In physics, such cheat sheets are also called local hidden variables.

The fact that quantum strategies can beat classical strategies in this game, however, implies that nature does not work that way! There are no classical cheat sheets, but nature is inherently quantum. Many experiments of ever increasing accuracy have been performed that verify that Alice and Bob can indeed achieve a higher winning probability in the CHSH game than the classical world would allow. Recently, an experiment has even proved this, by closing all possible loopholes (caused by experimental imperfections)[Hen+15]. This tells us that the world is not classical, but we need more sophisticated tools to describe it - such as quantum mechanics. It also means that when trying to build the ultimate computing and communication devices, we should make full use of what nature allows and go quantum.

We will later see how to use this simple game to verify the presence of entanglement, test unknown quantum devices, and even create secure encryption keys.

2.5 The monogamy of entanglement

Let's get back to the property mentioned in the very beginning of this lecture: that entanglement is monogamous. We know that two systems A and B can be in a joint pure state that is entangled, such as the "maximally entangled" EPR pair $|\text{EPR}\rangle_{AB}$. All our examples, however, had to do with entanglement between two systems A and B. But what about a third system, call it C for Charlie? Of course we could always consider three EPR pairs, $|\text{EPR}\rangle_{AB}$, $|\text{EPR}\rangle_{BC}$ and $|\text{EPR}\rangle_{AC}$. If this is the state of the three systems however we don't really want to talk about tripartite entanglement, because the correlations are always between any two of the three parties. Is it possible to create a joint state $|\Psi\rangle_{ABC}$ in which the strong correlations of the EPR pair are shared simultaneously between all three systems?

Let's first argue that, if we require that A and B are strictly in an EPR pair, then it is impossible for C to share any correlation with the qubits that form the EPR pair.

■ **Example 2.5.1** Let $\rho_{AB} = |\Psi\rangle\langle\Psi|_{AB}$. Then ρ_{AB} is pure, and in particular its only nonzero eigenvalue is $\lambda_1 = 1$. Thus by Uhlmann's theorem any purification of ρ_{AB} must have the form $\rho_{ABC} = |\Psi\rangle\langle\Psi|_{AB} \otimes |\Phi\rangle\langle\Phi|_C$ for an arbitrary state $|\Phi\rangle_C$ of system C. But this is a pure state with Schmidt rank across $AB : C$ equal to 1: it is not entangled! In fact you can see that the same consequence would hold as soon as AB is required to be in a pure state. In our example, you can further compute that $\rho_{AC} = \frac{\mathbb{I}}{2} \otimes \rho_C$, meaning that not only C is uncorrelated with A, but from the point of view of C A looks maximally mixed, i.e. it completely random. The same holds for ρ_{BC}. ■

2.5.1 Quantifying monogamy

The previous example demonstrates monogamy of the maximally entangled EPR pair. What about more general states, could they demonstrate entanglement across three different parties? This is possible to some extent, as is shown by the example of the GHZ state $|\Psi\rangle_{ABC} = \frac{1}{\sqrt{2}}(|000\rangle + |111\rangle)$. But the correlations in that state are weaker than those of a maximally entangled state. How do we make this statement precise?

One possibility is to use so-called *entanglement measures* $E(A : B)$. An entanglement measure is any function of bipartite density matrices that satisfies certain desirable properties. We already saw such a measure, the Schmidt rank; however it only applies to pure bipartite states. For states that are not pure the situation is much more complicated, and there is no standard entanglement measure that satisfies all the properties that we would like. Among these properties, there is one which expresses monogamy as follows: for any tripartite density matrix ρ_{ABC} it requires that

$$E(A : B) + E(A : C) \leq E(A : BC). \tag{2.43}$$

One way to interpret this inequality is that, whatever the *total* entanglement that A has with B and C (right-hand side), this entanglement must split additively between entanglement with B and with C (left-hand side). You may think this is obvious — but in fact very few entanglement measures are known to satisfy the monogamy inequality (2.43)!

2.5.2 A three-player CHSH game

Another, more intuitive way of measuring monogamy is through the use of nonlocal games, such as the CHSH game that we discussed in Section 2.4. First consider a three-player variant of this game where Alice would be required to successfully play the CHSH game simultaneously with two different partners, Bob and Charlie. That is, Alice would be sent a random x, Bob a random y and Charlie a random z; they would have to provide answers a, b and c respectively such that $xy = a + b \mod 2$ *and* $xz = a + c \mod 2$. Can they do it? The fact, discussed in Example 2.5.1, that the EPR pair has no entangled extension to three parties should give you a hint that things are going to be difficult for Alice!

In fact it is possible to make an even stronger statement. Consider now the following three-player variant of the CHSH game:

- The referee selects two of the three players at random, and sends each of them the message "You've been selected!".
- The referee plays the CHSH game with the selected players, sending each of them a random question and checking their answers for the CHSH condition. The third player is completely ignored.

Now, what do you think is the players' maximum success probability in this game? For the case of classical players the answer should be clear: $3/4$. Indeed, there is nothing more or less they can do in this variant than in the original two-player CHSH. (Make sure you are convinced of this fact. What is an optimal strategy for the three players?)

What about quantum players? Can they win with probability $\cos^2(\pi/8)$? Why not? Let's think of a possible extension of the two-player strategy we saw in Section 2.4.1. First of all we need the

three players, Alice, Bob and Charlie, to decide on an entangled state to share. Given they know two of them are going to be asked to play CHSH, it is natural to set things up with three EPR pairs, one between Alice and Bob, another between Bob and Charlie, and the third between Alice and Charlie.

Now the game starts, and two players are told they are to play the game. However, the crucial point to observe is that each of the selected players is not told with whom they are to play the game! So, for instance Alice will know she has been selected, but will not be told who is the other lucky winner — Bob or Charlie. Which EPR pair is she going to use to implement her strategy?

It turns out there is no answer to this question: Alice is stuck! Although we won't do it here, it is possible to show that the optimal winning probability in the three-player CHSH game described above, for quantum players, is no larger than the classical optimum: $3/4$. (See [Ton09] for more details if you are interested in seeing how to show this.) This is a powerful demonstration of monogamy of entanglement, showing in particular that there is no nice extension of the EPR pair to a tripartite state — at least not one that allows any two of them to win the CHSH game! We will return to a similar manifestation of monogamy by analyzing a "tripartite guessing game" next week.

Acknowledgements

 The lecture notes are written by Nelly Ng, Thomas Vidick and Stephanie Wehner. We thank David Elkouss, Kenneth Goodenough, Jonas Helsen, Jérémy Ribeiro, and Charles Xu for proofreading.

Important identities for calculations

Purification of states

Given any density matrix diagonalized as $\rho_A = \sum_i \lambda_i |\phi_i\rangle\langle\phi_i|_A$, a purification of A is

$$|\Psi\rangle_{AB} = \sum_i \sqrt{\lambda_i} |\phi_i\rangle_A |w_i\rangle_B, \tag{2.44}$$

for any set of orthonormal vectors $\{|w_i\rangle_B\}_i$.

Schmidt decomposition of bipartite pure states

Any bipartite pure state $|\Psi\rangle_{AB}$ can be written into the form

$$|\Psi\rangle_{AB} = \sum_{i=1}^{d} \sqrt{\lambda_i} |a_i\rangle_A |b_i\rangle_B, \tag{2.45}$$

where $\{|a_i\rangle_A\}_i, \{|b_i\rangle_B\}_i$ are orthonormal vector sets, and $\sum_{i=1}^{d} \lambda_i = 1$.

CHSH game winning probability

Consider Alice and Bob playing in a game, where questions $x, y \in \{0,1\}$ are sent to them, and they respond with answers $a, b \in \{0,1\}$ respectively. Alice and Bob win the game if $a + b \pmod 2 = x \cdot y$. The winning probability is given by

$$p_{\text{win}}^{\text{CHSH}} = \frac{1}{4} \sum_{x,y \in \{0,1\}} \sum_{\substack{a,b \\ a+b \\ \text{mod } 2 = x \cdot y}} p(a,b|x,y) \ . \tag{2.46}$$

For any classical strategy, $p_{\text{win}}^{\text{CHSH}} = \frac{3}{4}$.
If Alice and Bob shares an EPR pair, then $p_{\text{win}}^{\text{CHSH}} = \cos^2 \frac{\pi}{8} \approx 0.85$.

Bibliography

[Bel64] John S Bell. *On the Einstein Podolsky Rosen Paradox*. 1964 (cited on page 9).

[Bru+14] Nicolas Brunner et al. "Bell nonlocality". In: *Reviews of Modern Physics* 86.2 (2014), page 419 (cited on page 9).

[Hen+15] Bas Hensen et al. "Loophole-free Bell inequality violation using electron spins separated by 1.3 kilometres". In: *Nature* 526.7575 (2015), pages 682–686 (cited on page 12).

[NC01] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2001 (cited on page 6).

[Ton09] Ben Toner. "Monogamy of non-local quantum correlations". In: *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*. Volume 465. 2101. The Royal Society. 2009, pages 59–69 (cited on page 14).

Lecture Notes

edX Quantum Cryptography: Week 3

Contents

We have seen in Week 1 an example of communication between Alice and Bob, where the transmitted message is hidden from any eavesdropper Eve. There, we have seen the importance of using a large key K shared between Alice and Bob, but looks completely random from Eve's perspective. In the next few lectures, we will concern ourselves with how to establish such a key.

In this week, we will first learn about ways to quantify quantum information, which will be crucial in formulating what does it mean to be secure in cryptographic protocols.

3.1 When are two quantum states almost the same?

It will be important for us to have some notion of what it means to approximately produce a particular quantum state.

3.1.1 Trace distance

One measure of closeness that is of extreme importance in quantum cryptography, and also in the design of quantum circuits is the trace distance. Let us suppose, we would like to implement a protocol or algorithm that produces state ρ_{ideal}. Unfortunately, due to imperfections, our protocol produces the state ρ_{real}. If we now use this protocol or algorithm as a subroutine in a much larger protocol or computation, how is this larger protocol affected if we can only make ρ_{real} instead of ρ_{ideal}?

Intuitively, it is clear that if ρ_{real} and ρ_{ideal} are nearly impossible to distinguish, then it should not matter much in the large protocol which one we use. We would thus like a distance measure that is directly related to how well we can distinguish the two states. To this end, let us suppose that we really don't know whether we have the real or ideal state. Imagine that we are given ρ_{real} and ρ_{ideal} each with probability $1/2$, and we are challenged to distinguish them. To this end, we can perform a measurement using some operators M_{real} and $M_{\text{ideal}} = \mathbb{I} - M_{\text{real}}$. The probability of distinguishing the two states is then

$$p_{\text{succ}} = \frac{1}{2}\operatorname{tr}\left[M_{\text{real}}\rho_{\text{real}}\right] + \frac{1}{2}\operatorname{tr}\left[M_{\text{ideal}}\rho_{\text{ideal}}\right] = \frac{1}{2} + \frac{1}{2}\operatorname{tr}\left[M_{\text{real}}\left(\rho_{\text{real}} - \rho_{\text{ideal}}\right)\right] . \tag{3.1}$$

To find the best measurement, we can optimize the term M_{real} above over all measurement operators. We know (see Week 1 lecture notes, section on POVMs) that $0 \leq M_{\text{real}} \leq \mathbb{I}$, i.e. M_{real}'s eigenvalues all lie between 0 and 1. Thus the maximum success probability is given by

$$p_{\text{succ}}^{\max} = \frac{1}{2} + \frac{1}{2}\max_{0\leq M\leq\mathbb{I}} \operatorname{tr}\left[M\left(\rho_{\text{real}} - \rho_{\text{ideal}}\right)\right] . \tag{3.2}$$

What is, then, the operator M that would maximize the trace quantity $\operatorname{tr}\left[M\left(\rho_{\text{real}} - \rho_{\text{ideal}}\right)\right]$? This question has been analyzed in [Hel76], and the optimal M is the projector onto the positive eigenspace of $\rho_{\text{real}} - \rho_{\text{ideal}}$. More concretely, consider the diagonalized form of the linear operator $\rho_{\text{real}} - \rho_{\text{ideal}}$, and denote this diagonal matrix as $D = \sum_i d_i |d_i\rangle\langle d_i|$. Furthermore, denote the set $S_+ = \{j | d_j > 0\}$. The optimal M is then given by

$$M_{\text{opt}} = \sum_{j\in S_+} |d_j\rangle\langle d_j|. \tag{3.3}$$

It turns out the the trace distance precisely captures this idea of distinguishing states.

Definition 3.1.1 — Trace distance. The *trace distance* between two quantum states ρ_{real} and ρ_{ideal} is given by

$$D(\rho_{\text{real}}, \rho_{\text{ideal}}) = \max_{0\leq M\leq\mathbb{I}} \operatorname{tr}\left[M\left(\rho_{\text{real}} - \rho_{\text{ideal}}\right)\right] . \tag{3.4}$$

The trace distance can also be written as

$$D(\rho_{\text{real}}, \rho_{\text{ideal}}) = \frac{1}{2} \text{tr}\left[\sqrt{A^\dagger A}\right] , \tag{3.5}$$

where $A = \rho_{\text{real}} - \rho_{\text{ideal}}$.

In the literature, you will also see the trace distance written using the following notation

$$D(\rho_{\text{real}}, \rho_{\text{ideal}}) = \frac{1}{2} \|\rho_{\text{real}} - \rho_{\text{ideal}}\|_{\text{tr}} = \frac{1}{2} \|\rho_{\text{real}} - \rho_{\text{ideal}}\|_1 . \tag{3.6}$$

If two states are close in trace distance, then there exists no measurement - no process in the universe - that can tell them apart very well. It also means that if we use a subroutine that makes ρ_{real} instead of ρ_{ideal} and the two are close in trace distance, then we can safely conclude that also the surrounding larger protocol cannot see much difference. Otherwise, we could use the large protocol to tell the two states apart, but we know this cannot be.

Definition 3.1.2 — Closeness in terms of trace distance. Two quantum states ρ and σ are ε-close, if $D(\rho, \sigma) \leq \varepsilon$. We also write this as $\rho \approx_\varepsilon \sigma$.

Proposition 3.1.1 The trace distance is a metric, that is, a proper distance measure that corresponds to our intuitive notions of distance. We have the following properties for all states ρ, σ, τ:

1. Non-negative: $D(\rho, \sigma) \geq 0$, where equality is achieved if and only if $\rho = \sigma$.
2. Symmetric: $D(\rho, \sigma) = D(\sigma, \rho)$.
3. Triangle inequality: $D(\rho, \sigma) \leq D(\rho, \tau) + D(\tau, \sigma)$.
4. Convexity: $D(\sum_i p_i \rho_i, \sigma) \leq \sum_i p_i D(\rho_i, \sigma)$.

■ **Example 3.1.1** Consider $\rho_1 = |0\rangle\langle 0|$ and $\rho_2 = |+\rangle\langle +|$. Firstly, calculate

$$\rho_1 - \rho_2 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} - \frac{1}{2} \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 & -1 \\ -1 & -1 \end{pmatrix} . \tag{3.7}$$

Therefore, the trace distance is equal to

$$D(\rho_1, \rho_2) = \frac{1}{2} \cdot \frac{1}{2} \text{tr} \sqrt{\begin{pmatrix} 1 & -1 \\ -1 & -1 \end{pmatrix}^2} = \frac{1}{2} \cdot \frac{1}{2} \text{tr} \sqrt{\begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}} = \frac{1}{\sqrt{2}} . \tag{3.8}$$

Another way to do so is to first consider the diagonalization of $\rho_1 - \rho_2$, which can be done by first calculating its eigenvalues, solving the following equation:

$$\det \begin{pmatrix} \frac{1}{2} - \lambda & -\frac{1}{2} \\ -\frac{1}{2} & -\frac{1}{2} - \lambda \end{pmatrix} = 0. \tag{3.9}$$

The solutions are given by $\lambda = \pm \frac{1}{\sqrt{2}}$. One can also find the eigenvector $|e_+\rangle = (x \quad y)^T$ corresponding to $\lambda = \frac{1}{\sqrt{2}}$,

$$\frac{1}{2} \begin{pmatrix} 1 & -1 \\ -1 & -1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} x \\ y \end{pmatrix} \quad \Longrightarrow \quad \frac{x}{y} = \frac{-1}{\sqrt{2} - 1} . \tag{3.10}$$

On the other hand, normalization condition gives $x^2 + y^2 = 1$, and the solution is found to be

$$x = \cos \frac{\pi}{8}, \quad y = \sin \frac{\pi}{8} . \tag{3.11}$$

The optimal measurement operator that distinguishes ρ_1, ρ_2 is then given by $M_{\text{opt}} = |e_+\rangle\langle e_+|$, while

$$\text{tr}\left[M_{\text{opt}} (\rho_1 - \rho_2)\right] = \frac{1}{\sqrt{2}} . \tag{3.12}$$

■

Since states which are $\varepsilon-$close to each other cannot be distinguished well, it will later be helpful to have the notion of a set of states which are all $\varepsilon-$close to a particular state ρ. This is often called the $\varepsilon-$ball of ρ.

Definition 3.1.3 — $\varepsilon-$ball of ρ. Given any density matrix ρ, the *$\varepsilon-$ball of ρ* is defined as the set of all states ρ' which are $\varepsilon-$close to ρ in terms of trace distance, i.e.

$$\mathscr{B}^{\varepsilon}(\rho) := \{\rho' \mid \rho' \geq 0, \mathrm{tr}(\rho') = 1, D(\rho,\rho') \leq \varepsilon\}. \tag{3.13}$$

3.1.2 Fidelity

Although we have not seen this in the lectures, there is another common measure for closeness of states is known as the fidelity, which for pure states is directly related to their inner product.

Definition 3.1.4 — Fidelity. Given density matrices ρ_1 and ρ_2, the *fidelity* between ρ_1 and ρ_2 is

$$F(\rho_1,\rho_2) = \mathrm{tr}\left[\sqrt{\sqrt{\rho_1}\rho_2\sqrt{\rho_1}}\right] . \tag{3.14}$$

For pure states $\rho_1 = |\Psi_1\rangle\langle\Psi_1|$ and $\rho_2 = |\Psi_2\rangle\langle\Psi_2|$ the fidelity takes on a simplified form:

$$F(\rho_1,\rho_2) = |\langle\Psi_1|\Psi_2\rangle| . \tag{3.15}$$

If only one of the states $\rho_1 = |\Psi_1\rangle\langle\Psi_1|$ is pure, we have

$$F(\rho_1,\rho_2) = \sqrt{\langle\Psi_1|\rho_2|\Psi_1\rangle} . \tag{3.16}$$

Although the fidelity is not a metric (since $F(\rho_1,\rho_2) = 0$ does not imply that $\rho_1 = \rho_2$), it does have an intuitive interpretation, if we were to verify whether we managed to produce a desired target state $|\Psi\rangle$. Suppose that we want to build a machine that produces $|\Psi\rangle\langle\Psi|$, yet we are only able to produce some state ρ. Let us suppose we now measure ρ to check for success. We can do this (theoretically) by measuring

$$M_{\mathrm{succ}} = |\Psi\rangle\langle\Psi| , \tag{3.17}$$
$$M_{\mathrm{fail}} = \mathbb{I} - |\Psi\rangle\langle\Psi| . \tag{3.18}$$

The success probability is directly related to the fidelity between the true output ρ and the target state $|\Psi\rangle$ as

$$\mathrm{tr}[M_{\mathrm{succ}}\rho] = \langle\Psi|\rho|\Psi\rangle = F(|\Psi\rangle,\rho)^2 . \tag{3.19}$$

It is interesting to note that another way to write the fidelity is as

$$\max_{|\rho_{AP}\rangle,|\sigma_{AP}\rangle} |\langle\rho_{AP}|\sigma_{AP}\rangle| , \tag{3.20}$$

where $|\rho_{AP}\rangle$ and $|\sigma_{AP}\rangle$ are purifications of the states ρ_A and σ_A using a purifying system P.

Proposition 3.1.2 For any two quantum states ρ,σ, the fidelity satisfies the following properties

1. Between 0 and 1: $0 \leq F(\rho,\sigma) \leq 1$.
2. Symmetric: $F(\rho,\sigma) = F(\sigma,\rho)$.
3. Multiplicative under tensor product: $F(\rho_1\otimes\rho_2,\sigma_1\otimes\sigma_2) = F(\rho_1,\sigma_1) + F(\rho_2,\sigma_2)$.
4. Invariant under unitary operations: $F(\rho,\sigma) = F(U\rho U^\dagger, U\sigma U^\dagger)$.
5. Relation to trace distance: $1 - F(\rho,\sigma) \leq D(\rho,\sigma) \leq \sqrt{1-F^2(\rho,\sigma)}$. Conversely, we also have that $1 - D(\rho,\sigma) \leq F(\rho,\sigma) \leq \sqrt{1-D^2(\rho,\sigma)}$. This is known as the Fuchs-van de Graaf inequality [FV99].

3.2 Measuring uncertainty: the min-entropy

In many quantum protocols, we will be measuring quantum states and not get a key immediately. Instead, we will create a cq-state

$$\rho_{XE} = \sum_{x\in\{0,1\}^n} p_x |x\rangle\langle x|_X \otimes \rho_x^E \ , \tag{3.21}$$

where we have used the shorthand $p_x = \text{Prob}(X = x)$, and p_x is not uniform. Of course, we could consider the distance of this state to an ideal state ρ_{XE}^{ideal}, but it will typically be extremely large. Nevertheless, we could ask how useful the state ρ_{XE} for obtaining a key, for example, by performing some computation on the string X. This motivates us to try and find a measure of uncertainty about the classical string X.

3.2.1 The min-entropy

Let us first consider just the state

$$\rho_X = \sum_x p_x |x\rangle\langle x|_X \ . \tag{3.22}$$

Note that this means that we are effectively considering the probability distribution p_x over strings x. How could we measure the uncertainty inherent in ρ_X? When talking about communication, one very important measure is the von Neumann or Shannon entropy $H(X) = -\sum_x p_x \log p_x$. Is this quantity also a useful measure in the context of cryptography?

To think about this question, let us consider the following scenario: Suppose we have purchased a box (possibly from Eve!) which generates a string $x = x_1, \ldots, x_n$. If the string was uniformly random, then $p_x = 1/2^n$ and $H(X) = n$. If x is uncorrelated from Eve, then we could hope to use the string x as an encryption key for use in the one-time pad. Suppose now that while we are promised that x is uncorrelated from Eve, the distribution p_x is *not* uniform. However, we are guaranteed that the entropy is still $H(X) \approx n/2$, and n is very large. We know nothing else about the box. Would you still be willing to use x as an encryption key?

On first sight, the situation may not be so bad. After all, while the string does not have maximum entropy $H(X) = n$, it still has half as much entropy, which for very large n is after all still extremely large. Intuitively, this should mean that there is a lot of uncertainty for Eve, or does it?

Let us consider the following distribution:

$$p_x = \begin{cases} \frac{1}{2} & \text{for } x = 11\ldots1 \\ \frac{1}{2}\cdot\frac{1}{2^n-1} & \text{otherwise} \ . \end{cases} \tag{3.23}$$

Exercise 3.2.1 Show that the entropy for this distribution is $H(X) \approx n/2$. ∎

But is there a lot of uncertainty for Eve? Note that the probability that the box generates the string $x = 11\ldots1$ is $1/2$, independent of the length of the string! This means that whenever we use x as an ecryption key, Eve will be able to guess the key, and thus decrypt the message with probability $1/2$. Eve's probability of guessing is extremely large, even when we send a very large message.

We thus see that the von Neumann/Shannon entropy is not a good measure for cryptography. However, there exists an alternate entropy which is indeed useful for such purposes.

Definition 3.2.1 — Min-entropy. Given any probability distribution $\{p_x\}_x$, the *min-entropy* $\text{H}_{\min}$ is defined as $\text{H}_{\min}(X) = \text{H}_{\min}(\rho_X) = -\log\max_x p_x$.

In our example above, we see that $\mathrm{H}_{\min}(X) = -\log 1/2 = 1$. That is, the min-entropy is constant! Note that the min-entropy precisely captures our intuitve idea of what it means for Eve to be uncertain about x: Eve could guess the string with probability $1/2$. In general, we would all guess the most likely string, and the probability that we are correct is precisely $P_{\text{guess}}(X) = \max_x p_x$. The min-entropy thus has as very neat operational interpretation as

$$\mathrm{H}_{\min}(X) = -\log P_{\text{guess}}(X) \ . \tag{3.24}$$

R We may wonder why this was not also the right measure of uncertainty in the communication tasks we considered. Note that there we have always look at the case where we have states of the form $\rho^{\otimes n}$ where n is reasonably large. Following Shannon's line of thought and thinking of $i(x) := -\log p_x$ as the surprisal, that is, the information gained when we observe x, the Shannon entropy measured the *average* surprisal $H(X) = \sum_x p_x i(x)$. When doing cryptography, however, we are always interested in the worst case, not the average case. The min-entropy $\mathrm{H}_{\min}(X) = \min_x i(x)$ is precisely this smallest surprisal. Fig.3.1 shows the difference between these quantities, for a binary random variable.

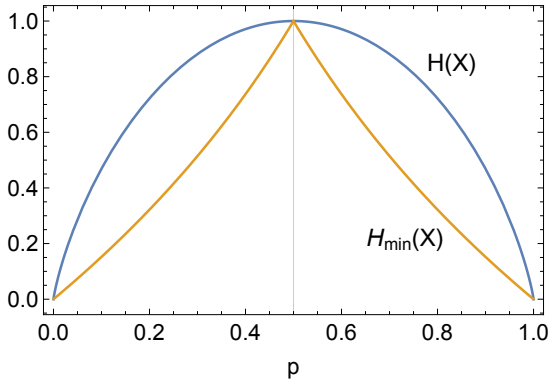

Figure 3.1: For a binary random variable $X = \{0,1\}$, the comparison between Shannon entropy $H(X)$ and its min-entropy $\mathrm{H}_{\min}(X)$.

Exercise 3.2.2 Show that the min-entropy satisfies the following bounds:

$$0 \leq \mathrm{H}_{\min}(X) \leq H(X) \leq \log |X|. \tag{3.25}$$

■

3.2.2 The conditional min-entropy

Can we also quantify the uncertainty about X *given* some extra quantum register E? It turns out that just like for the von Neumann entropy, the min-entropy has a conditional variant $\mathrm{H}_{\min}(X|E)$ developed in [Ren08]. The easiest way to think about the conditional min-entropy is in terms of the probability that Eve manages to guess X given access to her quantum register E. Note that we see from the cq-state in Eq. (3.21) that Eve has state ρ_x^E with probability p_x and her goal is to guess x by making a measurement on E. This is precisely the problem of distinguishing quantum states that we considered earlier.

Definition 3.2.2 — Conditional min-entropy. Consider a bipartite cq-state ρ_{XE} where X is classical. The *conditional min-entropy* $\mathrm{H}_{\min}(X|E)$ can be written as

$$\mathrm{H}_{\min}(X|E)_{\rho_{XE}} := -\log P_{\text{guess}}(X|E) \ , \tag{3.26}$$

where $P_{\text{guess}}(X|E)$ is the probability that Eve guesses x, maximized over all possible measurements

$$P_{\text{guess}}(X|E) := \max_{\{M_x\}_x} \sum_x p_x \operatorname{tr}\left[M_x \rho_x^E\right] , \tag{3.27}$$

where the maximization is taken over all POVMS $\{M_x \geq 0 \mid \sum_x M_x = \mathbb{I}\}$. In this context, E is also called *side information* about X. When it is clear from context, we omit the subscript ρ_{XE}, i.e. we write $\mathrm{H}_{\min}(X|E)_{\rho_{XE}} = \mathrm{H}_{\min}(X|E)$.

How could we ever hope to compute this quantity? When $x \in \{0,1\}$ takes on only two values, then it is easy to find the optimal measurement, and the guessing probability P_{guess} is directly related to the distinguishability of reduced states ρ_0^E and ρ_1^E, i.e. the trace distance $D(\rho_0^E, \rho_1^E)$. We shall see this in the following example.

■ **Example 3.2.1** Consider the state $\rho_{XE} = \frac{1}{2}|0\rangle\langle 0|_X \otimes |0\rangle\langle 0|_E + \frac{1}{2}|1\rangle\langle 1|_X \otimes |+\rangle\langle +|_E$. Then the conditional min-entropy $\mathrm{H}_{\min}(X|E) = -\log P_{\text{guess}}(X|E)$ where

$$\begin{aligned}
P_{\text{guess}}(X|E) &= \max_{\substack{M_1, M_2 \geq 0 \\ M_1 + M_2 = \mathbb{I}}} \left[\frac{1}{2}\operatorname{tr}(M_0|0\rangle\langle 0|_E) + \frac{1}{2}\operatorname{tr}(M_1|+\rangle\langle +|_E)\right] && (3.28)\\
&= \max_{0 \leq M \leq \mathbb{I}} \left[\frac{1}{2}\operatorname{tr}(M|0\rangle\langle 0|_E) + \frac{1}{2}\operatorname{tr}(|+\rangle\langle +|_E) - \frac{1}{2}\operatorname{tr}(M|+\rangle\langle +|_E)\right] && (3.29)\\
&= \frac{1}{2} + \frac{1}{2} \max_{0 \leq M \leq \mathbb{I}} \operatorname{tr}[M(|0\rangle\langle 0|_E - |+\rangle\langle +|_E)] && (3.30)\\
&= \frac{1}{2} + \frac{1}{2} D(|0\rangle\langle 0|_E, |+\rangle\langle +|_E). && (3.31)
\end{aligned}$$

■

However, if x can take more than two possible values, then it is in general difficult to compute $P_{\text{guess}}(X|E)$ by hand. Nevertheless, finding the optimal success probability is a so-called semi-definite program (SDP) and can be evaluate efficiently (in the dimension of the states ρ_x^E) using for example Matlab or Julia.

For any cq-state ρ_{XE} we have

$$0 \leq \mathrm{H}_{\min}(X|E) \leq \log|X| . \tag{3.32}$$

Note that the assumption that X is classical here is important: in particular, $\mathrm{H}_{\min}(X|E)$ can be negative if X is a genuine quantum register. Furthermore, we have

$$\mathrm{H}_{\min}(X|E) \geq \mathrm{H}_{\min}(X) - \log|E| . \tag{3.33}$$

A general quantum conditional min-entropy

In the fully general case, the system X as we have seen above is not necessarily classical, but can also be quantum (to make explicitly this difference, we use A to label such a quantum system). How should the conditional min-entropy $\mathrm{H}_{\min}(A|E)$ look like? To gain some intuition on how such a quantity should be defined, think of the guessing probability as a way of quantifying how close one may get *classically maximally correlated* with the classical system X, i.e. by guessing it correctly. Therefore, a quantum extension of this concept would be to, when only allowing to perform operations upon E, get as close as possible to the maximally entangled state between A and E.

Definition 3.2.3 — Quantum conditional min-entropy, (KRS09). Given any bipartite density matrix ρ_{AE}, with A having dimension $|A|$, the conditional min-entropy is

$$\mathrm{H}_{\min}(A|E) := -\log\left[|A| \cdot \mathrm{Dec}(A|E)\right], \tag{3.34}$$

$$\mathrm{Dec}(A|E) := \max_{\Lambda_{E\to A'}} F((\mathbb{I}_A \otimes \Lambda_{E\to A'})\rho_{AE}, |\Phi\rangle\langle\Phi|_{AA'})^2, \tag{3.35}$$

where $|\Phi\rangle_{AA'} := \frac{1}{\sqrt{|A|}}\sum_{i=1}^{|A|} |a_i\rangle_A \otimes |a_i\rangle_{A'}$ is the maximally entangled state between A and A', and the maximization is performed over all quantum channels Λ mapping system E to A'. The function F is the fidelity that we have seen in Def. 3.1.4.

An alternative way to express the conditional min-entropy is

$$\mathrm{H}_{\min}(A|E) := \max_{\sigma_B} \sup\{\lambda \in \mathbb{R} | \rho_{AB} \leq 2^{-\lambda}\mathbb{I}_A \otimes \sigma_B\}. \tag{3.36}$$

Smoothed min-entropy

As one has seen earlier in the discussion on trace distance, due to imperfections in a protocol or algorithm we often do not excactly produce the state ρ_{XE} that we want, rather, we can only manage to produce a state which is close, ρ'_{XE}, and we do not know the form of ρ'_{XE} (other than the fact that it is $\varepsilon-$close to ρ_{XE}). For this reason, it is usually more physically relevant to look at the *smoothed min-entropy*, which gives us the maximum value of $\mathrm{H}_{\min}(X|E)$ over all states $\rho'_{AE} \in \mathscr{B}^{\varepsilon}(\rho_{AE})$.

Definition 3.2.4 — Smoothed conditional min-entropy. Consider a bipartite cq-state ρ_{XE} where X is classical. The *smoothed conditional min-entropy* $\mathrm{H}^{\varepsilon}_{\min}(X|E)$ can be written as

$$\mathrm{H}^{\varepsilon}_{\min}(X|E)_\rho := \max_{\rho' \in \mathscr{B}^{\varepsilon}(\rho)} \mathrm{H}_{\min}(X|E)_{\rho'}. \tag{3.37}$$

3.3 What it means to be ignorant

Before establishing keys, let us be precise about what we actually want to achieve. We have already sketched before that we desire the keys to be picked from a uniformly random distribution, and Eve to be uncorrelated. Classically, we could say that this should mean that the probability of selecting any n-bit key k is $\mathrm{Prob}(K=k) = p(k) = 1/2^n$, and the key k is independent of some classical information, denoted by e, that the eavesdropper may have gathered. That is, $p(k) = p(k|e)$.

Clearly, it makes no sense to talk about some probability distribution over classical keys k conditioned on classical strings e in the quantum case. After all, Eve may have gathered quantum information about the key! That is, the state between the register holding the key, let us call it K, and the register of Eve, let us call it E, is a cq-state

$$\rho_{KE} = \sum_k p_k |k\rangle\langle k| \otimes \rho_k^E \ . \tag{3.38}$$

This implies that depending on the key k, Eve has a quantum state ρ_k^E that she may measure to gain some information about the key. Ideally, the fact that Eve knows nothing can be expressed in the following definition, which we refer to as ignorance about the key.

To motivate the definition of ignorance, let us first consider a few examples, where for simplicity we consider just a single bit of key. Our examples can however be easily extended to arbitrary many keys, and you're encouraged to check.

■ **Example 3.3.1** First, let us consider the state

$$\rho_{KE} = \frac{1}{2}\sum_{k\in\{0,1\}} |k\rangle\langle k|_K \otimes |k\rangle\langle k|_E \ . \tag{3.39}$$

Clearly, we have $\rho_K = \mathrm{tr}_E(\rho_{KE}) = \mathbb{I}_K/2$. That is the key is uniform. But clearly Eve knows everything about the key: whenever K is in the state $|k\rangle\langle k|$, then so is E! You may see the information that Eve has as simply a classical piece of paper that has an exact copy of k. States of the form above are also called classically maximally correlated states. Both systems are diagonal in the standard basis, and the both systems are prepared precisely in the same state $|k\rangle\langle k|$ with some probability. ∎

∎ **Example 3.3.2** Let us now consider the state $\rho_{KE} = |0\rangle\langle 0|_K \otimes \rho_E$. It sure appears Eve is uncorrelated. However, ρ_K is certainly not uniform. In fact, the only possible key is $k = 0$, so it is indeed easy to guess the key for anyone. ∎

∎ **Example 3.3.3** Consider the maximally entangled state

$$\rho_{KE} = |\Psi\rangle\langle\Psi|_{KE}$$

between K and E, that is, $|\Psi\rangle_{KE} = (|0\rangle_K|0\rangle_E + |1\rangle_K + |1\rangle_E)/\sqrt{2}$. As you have calculated before, we have $\rho_K = \mathrm{tr}_E(\rho_{KE}) = \mathbb{I}/2$. That is, the key X is uniform. But is it uncorrelated? Clearly not, no matter what basis we measure K in, there always exists a corresponding measurement on E that yields the same outcome. This is because for all unitaries U, we have

$$\begin{aligned}
U_K \otimes U_E^*|\Psi\rangle_{KE} &= (U_K \otimes \mathbb{I}_E)(\mathbb{I}_K \otimes U_E^*)|\Psi\rangle_{KE} && (3.40)\\
&= (U_K \otimes \mathbb{I}_E)((U_K^*)^T \otimes \mathbb{I}_E)|\Psi\rangle_{KE} && (3.41)\\
&= (U_K \otimes \mathbb{I}_E)(U_K^\dagger \otimes \mathbb{I}_E)|\Psi\rangle_{KE} && (3.42)\\
&= (U_K U_K^\dagger \otimes \mathbb{I}_E)|\Psi\rangle_{KE} && (3.43)\\
&= |\Psi\rangle_{KE}, && (3.44)
\end{aligned}$$

where in the second equality, we have made used of a special property that holds for $|\Psi\rangle_{KE}$: for any U, $(\mathbb{I}_K \otimes U_E)|\Psi\rangle_{KE} = (U_K^T \otimes \mathbb{I}_E)|\Psi\rangle_{KE}$. Therefore, the corresponding measurement on E is simply to measure in the basis defined by U_E^* (i.e. the basis in which U_E^* is diagonalized). ∎

Therefore, we conclude that an eavesdropper Eve is ignorant of a key if and only if the following conditions hold.

Definition 3.3.1 — Ignorant. Consider the joint cq-state ρ_{KE} of an n-bit key K and the eavesdropper Eve, E. Eve is *ignorant* about the key K if and only if

$$\rho_{KE} = \frac{1}{2^n}\mathbb{I}_K \otimes \rho_E. \tag{3.45}$$

That is, the key is uniform *and* uncorrelated from Eve.

In any actual implementation, we can never hope to attain the perfection as given by the state in Eq. (3.45). However, we can hope to get close to such an ideal state, motivating the following definition.

Definition 3.3.2 — Almost ignorant. Consider the joint cq-state ρ_{KE}^{real} of an n-bit key K and the eavesdropper Eve, E. Eve is *almost ignorant* about the key K if and only if

$$D\left(\rho_{KE}^{\text{real}}, \rho_{KE}^{\text{ideal}}\right) \leq \varepsilon\ , \tag{3.46}$$

where $\rho_{KE}^{\text{ideal}} = \frac{1}{2^n}\mathbb{I}_K \otimes \rho_E$.

Why would this be a good definition? Recall that the trace distance measures exactly how well we can distinguish two scenarios. We saw that if two states are ε-close in trace distance, then no measurement can tell them apart with an advantage more than $\varepsilon/2$, i.e. if we were given one of the

two states with equal probability, *any measurement* allowed by quantum mechanics would only tell them apart with probability $1/2+\varepsilon/2$. This is an advantage of at most $\varepsilon/2$ over a random guess, which would be correct with probability $1/2$.

This has important consequences if we want to later use the key in another protocol, for example, in an an encryption protocol like the one-time pad. Recall from Week 2 lecture notes that an encryption scheme is secret/secure if and only if for all prior distributions over the messages $p(m)$, and for all messages m, we should have $p(m) = p(m|c)$, where c denotes the ciphertext. Such a secrecy can be achieved using the one-time pad, if Eve is completely ignorant about the key. You may think of the one-time pad scheme as a type of measurement to distinguish ρ_{KE}^{ideal} and ρ_{KE}^{real}. If this protocol would behave very differently if we use the ρ_{KE}^{real} instead of the ideal ρ_{KE}^{ideal}, then this would give us a means to distinguish the two states very well. But this is precisely ruled out if the states are close in trace distance!

In conclusion, if $D\left(\rho_{KE}^{\text{real}}, \rho_{KE}^{\text{ideal}}\right) \leq \varepsilon$, while ρ_{KE}^{ideal} leads to the probabilty distribution $p(m) = p(m|c)$, then we should also have that when using the real state ρ_{KE}^{real}, $p(m) \approx_\varepsilon p(m|c)$ should hold. This means that in the analysis of any subsequent protocol we can assume that we have the ideal key, at the expense of only a very small error ε.

3.4 Uncertainty principles: a bipartite guessing game

In this section, we first see how to construct a simple guessing game that allows us to prove security against an eavesdropper Eve who can prepare quantum states, but who otherwise stores and processes only classical information. The crucial property of quantum mechanics which allows us to make this security proof is called the *uncertainty principle*. Such a principle tells us how well Eve can or cannot predict the outcomes of two incompatible measurements on Alice's state.

As a warmup, let us first consider the case where Eve only has classical memory. That is, she might make measurements on the qubits during the transmission, but she cannot keep any entanglement herself. This is effectively equivalent to Eve actually preparing Alice's qubits herself, and can be analyzed in the form of a guessing game defined below:

Definition 3.4.1 — Guessing game - Alice and Eve. Suppose Alice and Eve play the following game:

1. Eve prepares a qubit ρ_A and sends it to Alice.
2. Alice chooses a random bit $\Theta \in \{0,1\}$.
3. If $\Theta = 0$, then Alice measures ρ_A in the standard basis; if $\Theta = 1$, then she measures in the Hadamard basis.
4. Alice obtains and records a measurement outcome $X \in \{0,1\}$.
5. Alice announces Θ.
6. Eve wins if she can guess the bit X.

How may we make sure that Eve cannot fully predict Alice's measurement outcome X? As a simple example, let us return to Example 3.3.2 where the joint state between Alice and Eve is

$$\rho_{AE} = |0\rangle\langle 0|_A \otimes \rho_E, \tag{3.47}$$

where Alice measures system A either in the standard or Hadamard basis in order to obtain the key K. If Alice measures in the standard basis, Eve can always predict the outcome perfectly. However, if Alice measures in the Hadamard basis, Eve can only make a random guess, since by measuring Alice obtains outcome $|+\rangle$ and $|-\rangle$ each with probability $\frac{1}{2}$!

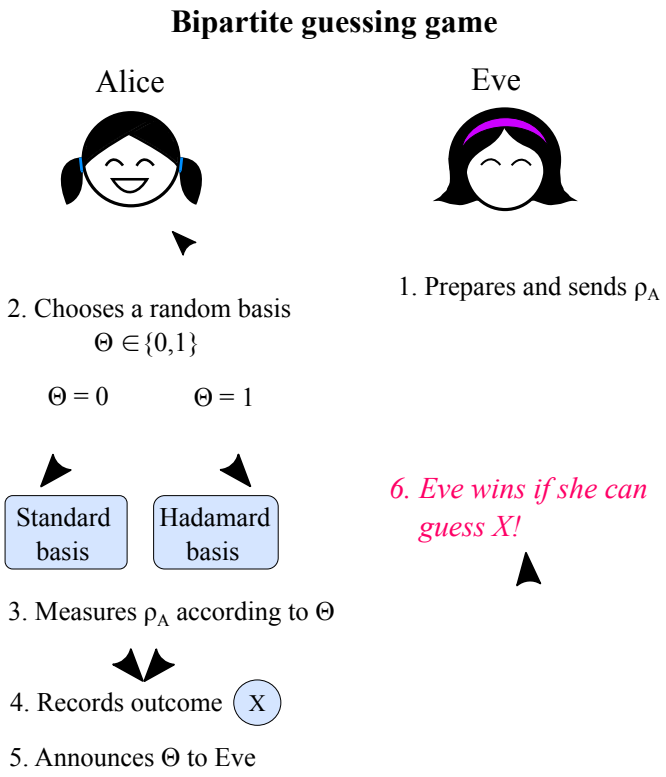

Figure 3.2: The guessing game between Alice and Eve, where Eve prepares a quantum state and sends it to Alice, who choses randomly to measure in the standard basis or in the Hadamard basis. Eve then tries to guess Alice's measurement outcome, given the basis she chosen.

To see why this captures the essence of the uncertainty principle, note that if the measurements are incompatible, then there exists no state ρ_A that Eve can prepare, that would allow her to guess the outcomes for both choices of measurements with certainty. Uncertainty can thus be quantified by a bound on the average probability that Eve correctly guesses X:

$$P_{\text{guess}}(X|\Theta) = p(\Theta=0)\cdot P_{\text{guess}}(X|\Theta=0) + p(\Theta=1)\cdot P_{\text{guess}}(X|\Theta=1) \tag{3.48}$$

$$= \frac{1}{2}\cdot\left[P_{\text{guess}}(X|\Theta=0) + P_{\text{guess}}(X|\Theta=1)\right] \leq c, \tag{3.49}$$

where the second equality holds if Alice chooses her measurement basis Θ at random, namely with uniform probability $\frac{1}{2}$ for each option. In the case where Eve holds no additional information except for the basis where Alice has performed the measurement, the quantity c can be shown to be strictly less than 1.

To see why this is the case, suppose that Eve aims to correctly guess X all of the time. In particular, she wants to guess X correctly always, regardless of whether $\Theta = 0$ or $\Theta = 1$. This means that she requires, in particular, that $P_{\text{guess}}(X|\Theta=0) = 1$, Eve should prepare a state that will always produce a deterministic outcome when Alice measures in the standard basis. In earlier weeks, we have seen that for this to happen, Eve can for example send the state $|0\rangle\langle 0|_A$, where Alice, upon measuring in the standard basis, will always produce $X = 0$. However, if Eve has used the strategy of preparing $|0\rangle\langle 0|_A$, what happens when Alice now measured in the Hadamard basis

instead? We can calculate the probability

$$P_{\text{guess}}(X|\Theta=1) = \max\left[\, p(X=0|\Theta=1), p(X=1|\Theta=1)\,\right] \tag{3.50}$$

$$= \max\left[\, \text{tr}\left(|+\rangle\langle+||0\rangle\langle 0|\right), \text{tr}\left(|-\rangle\langle-||0\rangle\langle 0|\right)\,\right] = \frac{1}{2}. \tag{3.51}$$

Therefore, if Eve uses this strategy of preparing $\rho_A = |0\rangle\langle 0|_A$ in order to guess Alice's outcome X, then whenever $\Theta = 1$, this corresponds only to a random guess! What's important in this protocol is that since Eve does not know beforehand what basis Alice will choose to measure in, she has to prepare a state that will maximize her guessing probability in *both* cases of Alice measuring in the standard basis, and also the Hadamard basis. We have seen from the above example that this guessing probability can never be equal to 1.

Note that in order for Eve to maximize the guessing probability $P_{\text{guess}}(X|\Theta)$ over ρ_A (without loss of generality one can consider the outcome to be $X = 0$),

$$P_{\text{guess}}(X|\Theta) = \frac{1}{2}\cdot\left[\text{tr}(\rho_A|0\rangle\langle 0| + \text{tr}(\rho_A|+\rangle\langle+|)\right] \tag{3.52}$$

$$= \frac{1}{2}\cdot\text{tr}\left[\rho_A(|0\rangle\langle 0| + |+\rangle\langle+|)\right]. \tag{3.53}$$

then she has to prepare ρ_A in the pure state corresponding to the eigenvector of $|0\rangle\langle 0| + |+\rangle\langle+|$ with the largest eigenvalue. Check for yourselves that the largest eigenvalue of this matrix is $\lambda_{\max} = 1 + \frac{1}{\sqrt{2}}$. Therefore, we have that $P_{\text{guess}}(X|\Theta) = \frac{1}{2} + \frac{1}{2\sqrt{2}} < 1$.

3.4.1 Analysis: winning probability of the guessing game

Let us first try to calculate Eve's guessing probability for the protocol in Def. 3.4.1. We have seen in previous lectures that any state can be written in its Bloch representation as $\rho_A = \frac{1}{2}(\mathbb{I} + v_x X + v_y Y + v_z Z)$, where the vector $\vec{v} = (v_x, v_y, v_z)$ is a 3-dimensional real vector. Therefore, one may calculate the following inner products using the Bloch representation:

$$\text{tr}(\rho_A|0\rangle\langle 0|) = \frac{1}{2}(1+v_z), \qquad \text{tr}(\rho_A|1\rangle\langle 1|) = \frac{1}{2}(1-v_z), \tag{3.54}$$

$$\text{tr}(\rho_A|+\rangle\langle+|) = \frac{1}{2}(1+v_x), \qquad \text{tr}(\rho_A|-\rangle\langle-|) = \frac{1}{2}(1-v_x). \tag{3.55}$$

On the other hand,

$$p_{\text{guess}}(X|\Theta) = \frac{1}{2}\max\{\text{tr}(\rho_A|0\rangle\langle 0|), \text{tr}(\rho_A|1\rangle\langle 1|)\} + \frac{1}{2}\max\{\text{tr}(\rho_A|+\rangle\langle+|), \text{tr}(\rho_A|-\rangle\langle-|)\}, \tag{3.56}$$

where we want this value maximized over all possible states ρ_A, since Eve is allowed to pick any arbitrary state. Since the maximizations of both expression are symmetric around $v_z = 0, v_x = 0$ respectively, we can without loss of generality consider only the case where $v_z, v_x \geq 0$. The expression in Eq. (3.56) then reduces to

$$p_{\text{guess}}(X|\Theta)_{\rho_A} = \frac{1}{2}\text{tr}\left[\rho_A\left(|0\rangle\langle 0| + |+\rangle\langle+|\right)\right] = \frac{1}{4}(2 + v_z + v_x), \qquad v_z^2 + v_x^2 \leq 1. \tag{3.57}$$

To maximize this expression over all states ρ_A implies maximizing Eq. (3.57) over the constraint $v_z^2 + v_x^2 \leq 1$, and the maximum happens only when $v_z^2 + v_x^2 = 1$ (this implies that ρ_A is pure for Eve's optimal strategy). Using the parametrization $v_z = \sin t, v_x = \cos t$ then gives us that

$$\max_t\ (\sin t + \cos t), \qquad \text{achieved when } \sin t = \cos t = \frac{1}{\sqrt{2}}. \tag{3.58}$$

Therefore, the probability Eve wins this game is $P_{\text{guess}}(X|\Theta)_{\rho_A} = 1/2 + 1/(2\sqrt{2}) \approx 0.85$.

In a more general scenario, Eve may even have classical information about ρ_A. This means that she is able to create an arbitrary cq-state $\rho_{AC} = \sum_c p_c \rho_c^A \otimes |c\rangle\langle c|_C$ according to some distribution $\{p_c\}_c$, and sends $\rho_A = \sum_c p_c \rho_c^A$ to Alice while keeping the classical label C. Let us further convince ourselves that any further classical information Eve holds about ρ_A does not help. Suppose that Eve can prepare any cq-state $\rho_{AC} = \sum_c p_c \rho_c^A \otimes |c\rangle\langle c|_C$, and sends ρ_A to Alice. The guessing probability further conditioned on C is given by

$$p_{\text{guess}}(X|\Theta C)_{\rho_{AC}} = \sum_c p_c p_{\text{guess}}(X|\Theta)_{\rho_c^A} \tag{3.59}$$

where we maximize over all possible $\{p_c, \rho_c^A\}_c$. But we have previously already shown the maximum possible value of $p_{\text{guess}}(X|\Theta)_{\rho_c^A}$, over all possible ρ_c^A! Therefore, Eq. (3.59) yields

$$p_{\text{guess}}(X|\Theta C)_{\rho_{AC}} = \frac{1}{2}\left(1 + \frac{1}{\sqrt{2}}\right) \approx 0.85. \tag{3.60}$$

This quantity $P_{\text{guess}}(X|\Theta C)$ now directly tells us about the min-entropy about this bit, since $H_{\min}(X|\Theta C) = -\log P_{\text{guess}}(X|\Theta C)$. That gives a value for min-entropy per bit, of $H_{\min}(X|\Theta C) = -\log P_{\text{guess}}(X|\Theta C) \approx 0.22$.

Next, let us give Eve a little more power. Suppose that not only can Eve prepare a state ρ_A for Alice to measure, she might create a larger state ρ_{AE}, possibly entangled, and send only ρ_A to Alice. Note that since we always allow Eve maximum information about everything, ρ_{AE} is always pure: Eve always holds the purification as well. We will thus simply assume that Eve can *prepare* pure states ρ_{AE} of which she sends qubit A to Alice.

Exercise 3.4.1 Show that if Eve can keep entanglement, that is the can prepare an arbitrary entangled state ρ_{AE} then she can guess X perfectly.
Hint: consider the case where Eve prepares the EPR pair. ■

When you complete the exercise you will discover that if Eve can be entangled with Alice's qubit, then she can guess perfectly.

Is there any hope for security (i.e. keeping X secret from Eve) at all then? The answer to this is yes: remember that entanglement is monogamous! In other words, if we want limit Eve's knowledge about Alice's measurement outcomes, then we need to use two aspects of quantum mechanics:

- Uncertainty: If Eve has no (or little) entanglement with Alice, then she cannot predict the outcomes of two incompatible measurements (very well). In particular, this means it is difficult for here to guess Alice's measurement outcomes, i.e., $P_{\text{guess}}(X|E\Theta) < 1$, or equivalently, $H_{\min}(X|E\Theta) > 0$.
- Entanglement: We need a means to ensure there actually is little entanglement between Alice and Eve. For this we can use the fact that entanglement is *monogamous*, that is, if we find a large amount of entanglement between Alice and Bob, then we know that Eve has very little entanglement with either Alice or Bob, and therefore the min-entropy should be large!

3.5 Extended uncertainty relation principles: A tripartite guessing game

In this section, in order to make use of the monogamous property of entanglement, we consider a direct extension of the guessing game as before, only this time we are given no guarantee about the entanglement (or absence thereof) between Alice and Eve. Instead, we have a third party Bob, whom Alice trusts. In particular, to show security against Eve, Alice and Bob may join forces to make an estimate of Eve's min-entropy. To do so, they need to perform an entanglement test

between Alice and Bob to ensure that by the monogamy of quantum entanglement, the entanglement between Alice and Eve is small. For this, let us consider the following tripartite guessing game.

Definition 3.5.1 — Tripartite guessing game - Alice, Bob and Eve. Suppose Alice plays against Bob and Eve in the following way:

1. Eve prepares a global state ρ_{ABE}, and sends qubits A and B to Alice and Bob respectively.
2. Alice chooses a random bit $\Theta \in \{0,1\}$.
3. If $\Theta = 0$, then Alice measures ρ_A in the standard basis; if $\Theta = 1$, then she measures in the Hadamard basis.
4. Alice obtains a measurement outcome $X \in \{0,1\}$ and records it.
5. Alice announces Θ to both Bob and Eve.
6. Given Θ, Bob measures ρ_B and makes a guess $\tilde{X}$. Likewise, Eve measures ρ_E and makes a guess X_E.
7. Bob and Eve win the game if $X_E = X = \tilde{X}$.

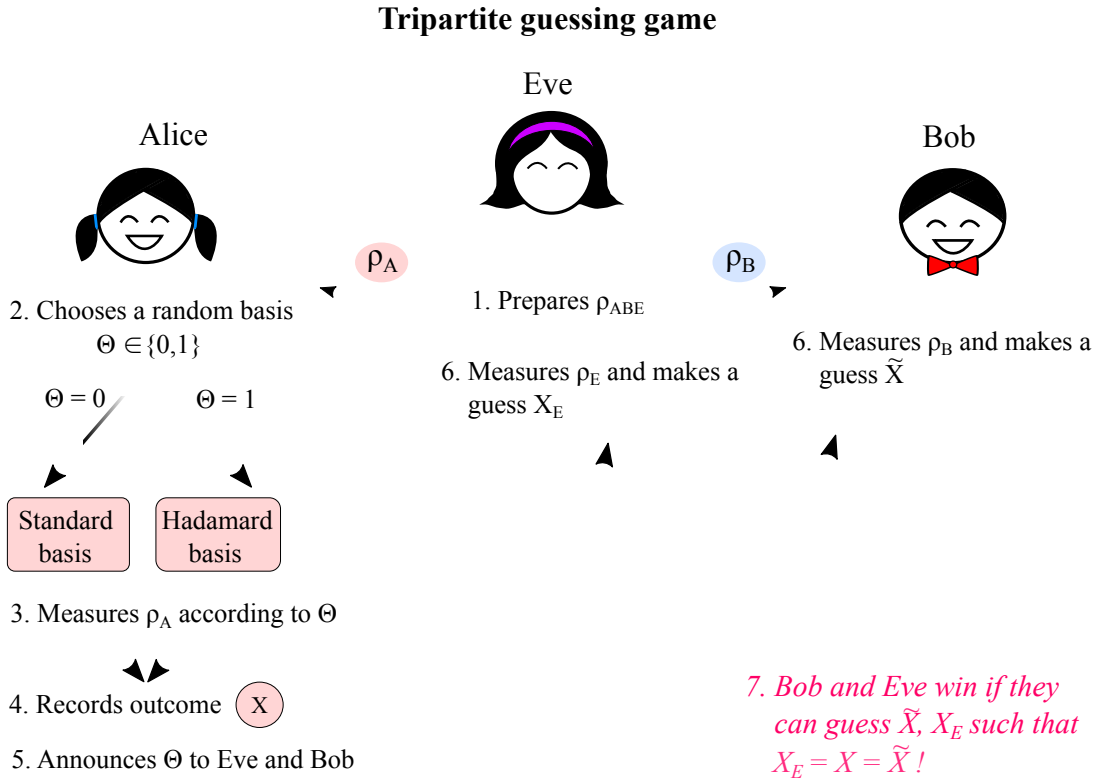

Figure 3.3: A tripartite guessing game where Eve gets to prepare the global state ρ_{ABE}. She send the qubits A and B to Alice and Bob respectively, where Alice measures randomly in either the standard or Hadamard basis. Bob and Eve both provide guesses $\tilde{X}, X_E$, and we say that they win the game if $X_E = X = \tilde{X}$.

Therefore, our goal will be to bound the probability that they all produce the same outcome,

averaged over the choice of basis, that is, that Bob and Eve wins the guessing game.

$$p_{\text{Tripartite}} = p\left(X = \tilde{X} = X_E\right) = \sum_{\Theta\in\{0,1\}} p_\Theta p(X = \tilde{X} = X_E|\Theta) \tag{3.61}$$

$$= \frac{1}{2}\sum_{\theta\in\{0,1\}} \text{tr}\left[\rho_{ABE}\left(\sum_{x\in\{0,1\}} |x\rangle\langle x|^A_\theta \otimes |x\rangle\langle x|^B_\theta \otimes M^E_{x|\theta}\right)\right] , \tag{3.62}$$

where we used superscripts A, B and E to denote the systems on which we perform the measurements, and $|x\rangle_\Theta$ to denote basis element x in the basis Θ. That is, $|0\rangle_0 = |0\rangle$, $|1\rangle_1 = |1\rangle$, and $|0\rangle_1 = |+\rangle$, $|1\rangle_1 = |-\rangle$. The probability above is the probability that they all give the same outcome x when measuring the state ρ_{ABE}. Of course, we don't know anything about the state ρ_{ABE} or the measurement $\{M^E_{x|\Theta}\}_x$ with outcomes x that Eve will perform on E depending on the basis Θ. We only know that this must be a quantum state, and Eve can only make measurements that are allowed by the laws of quantum mechanics. Since it is known that all POVMs can be realized as projective measurements using a potentially larger ancilla, and our all powerful Eve can hold the entire rest of the universe except Alice and Bob's labs, we can without loss of generality assume that Eve's measurements are projective. Given her access to a smaller space only makes things more difficult for Eve and in a security analysis we are always allowed to make the adversary more (but not less!) powerful.

3.5.1 Analysis: winning probability of the tripartite guessing game

How could we hope to analyze this situation? Previously when we considered a classical Eve, the solution was given by a simple eigenvalue problem, and if we would fix Eve's measurements then again we obtain an eigenvalue problem

$$\max_{\rho_{ABE}} \text{tr}\left[\rho_{ABE}\left(\frac{1}{2}\sum_\Theta \Pi_\Theta\right)\right] , \tag{3.63}$$

where

$$\Pi_\Theta = \sum_{x\in\{0,1\}} |x\rangle\langle x|^A_\Theta \otimes |x\rangle\langle x|^B_\Theta \otimes M^E_{x|\Theta} . \tag{3.64}$$

Now we are in some small amount of trouble given that we don't know $M^E_{x|\Theta}$ and malicious Eve will of course use the best possible measurements.

Two tools from linear algebra

To get around this dificulty, we will use two little linear algebra tricks which are proven in [Tom+13]. To write them down, let us first introduce a shorthand for the maximization problem above. In general, the *operator norm* of some operator O, can be written as

$$\|O\|_\infty = \max_\rho \text{tr}\left[\rho O\right] , \tag{3.65}$$

where the maximization is taken over all ρ such that $\text{tr}[\rho] \leq 1$. When, O is Hermitian, then we just maximize over all quantum states ρ, that is, ρ satisfying $\rho \geq 0$ and $\text{tr}[\rho] = 1$. Note that this means we can reduce the maximization problem above to studying

$$\left\|\frac{1}{2}\sum_{\theta\in\{0,1\}} \Pi_\theta\right\|_\infty , \tag{3.66}$$

for of course still partially unknown Π_θ. When talking about operators, people often omit the subscript ∞ and simply write $\|O\| = \|O\|_\infty$ as is also done in [Tom+13], and we will use this simpler notation from now on. Nevertheless, while cumbersome, one can establish the following two facts:

1. For any two projectors Π_0 and Π_1 [1], we have

$$\|\Pi_0 + \Pi_1\| \leq \max\{\|\Pi_0\|, \|\Pi_1\|\} + \|\Pi_0\Pi_1\|. \tag{3.67}$$

([Tom+13, Lemma 2])

2. If $\Pi_0 \leq P$ and $\Pi_1 \leq Q$ [2], then $\|\Pi_0\Pi_1\| \leq \|PQP\|$.

([Tom+13, Lemma 1])

Let us now use these two tricks to bound Eve's probability of winning. Using trick 1, we have that

$$\max_{M^E} \left\| \frac{1}{2} \sum_{\theta\in\{0,1\}} \Pi_\theta \right\|_\infty = \max_{M^E} \frac{1}{2} \left\| \sum_{\theta\in\{0,1\}} \Pi_\theta \right\|_\infty \tag{3.68}$$

$$\leq \frac{1}{2}\left(1 + \max_{M^E} \|\Pi_0\Pi_1\|\right), \tag{3.69}$$

where we have used that $\|\Pi_0\|, \|\Pi_1\| \leq 1$ for any measurements M^E that Eve could make in quantum mechanics (convince yourself that this is true!). It remains to analyze $\|\Pi_0\Pi_1\|$ for which we will use trick number two, for some smart choice of P and Q. Note that since all measurement operators $M^E_{x|\theta} \leq \mathbb{I}$ and also $|x\rangle\langle x|_\theta \leq \mathbb{I}$, we have that

$$\Pi_0 \leq \sum_{x\in\{0,1\}} |x\rangle\langle x|^A_0 \otimes |x\rangle\langle x|^B_0 \otimes \mathbb{I}^E =: P \tag{3.70}$$

$$\Pi_1 \leq \sum_{x\in\{0,1\}} |x\rangle\langle x|^A_1 \otimes \mathbb{I}^B \otimes M^E_{x|1} =: Q \tag{3.71}$$

Using the fact that $\langle x|y\rangle = 0$ if $x \neq y$ in the same basis, and that $\sum_y M^E_{y|1} = \mathbb{I}$ for any quantum measurement Eve may make, we thus have

$$PQP = \sum_{x,y,z} |x\rangle\langle x|^A_0 |y\rangle\langle y|^A_1 |z\rangle\langle z|^A_0 \otimes |x\rangle\langle x|^B_0 |z\rangle\langle z|^B_0 \otimes M^E_{y|1} \tag{3.72}$$

$$= \sum_{x,y} \frac{1}{2} |x\rangle\langle x|^A_0 \otimes |x\rangle\langle x|^B_0 \otimes M^E_{y|1} \tag{3.73}$$

$$= \frac{1}{2} \sum_x |x\rangle\langle x|^A_0 \otimes |x\rangle\langle x|^B_0 \otimes \sum_y M^E_{y|1} \tag{3.74}$$

$$= \frac{1}{2} \sum_x |x\rangle\langle x|^A_0 \otimes |x\rangle\langle x|^B_0 \otimes \mathbb{I}^E . \tag{3.75}$$

This gives $\|PQP\| \leq 1/2$. Using trick number two, and plugging into Eq. (3.69) we thus have that

$$p_{\text{Tripartite}} \leq \frac{1}{2}\left(1 + \frac{1}{\sqrt{2}}\right) = \frac{1}{2} + \frac{1}{2\sqrt{2}}, \tag{3.76}$$

which is again our familiar number from the much simpler guessing game, where Eve was all classical! Moreover, it can be shown using messy but not not more advanced mathematical tools that also when we consider collective attacks

$$p^{\text{n rounds}}_{\text{Tripartite}} \leq \left(\frac{1}{2} + \frac{1}{2\sqrt{2}}\right)^n , \tag{3.77}$$

[1] Recall that a projector Π is an operator such that $\Pi^2 = \Pi$.

[2] Recall that $A \leq B$ means that $B - A \geq 0$.

and she can achieve this bound by playing the optimal one round strategy!

We thus know that if the error rate is low, and Bob can reproduce a significant fraction $X = \tilde{X}$, then it is difficult for Eve to guess $X_E = X$ and hence her min-entropy must be large.

Acknowledgements

 The lecture notes are written by Stephanie Wehner, Nelly Ng, and Thomas Vidick. We thank David Elkouss, Jonas Helsen, Jérémy Ribeiro and Kenneth Goodenough for proofreading.

Important identities for calculations

Trace distance

$$D(\rho_{\text{real}}, \rho_{\text{ideal}}) := \max_{0 \le M \le \mathbb{I}} \text{tr}\left[M\left(\rho_{\text{real}} - \rho_{\text{ideal}}\right)\right] \tag{3.78}$$

$$= \frac{1}{2}\,\text{tr}\left[\sqrt{A^\dagger A}\right], \qquad A = \rho_{\text{real}} - \rho_{\text{ideal}}. \tag{3.79}$$

Properties:

1. $D(\rho,\rho') \ge 0$ with equality iff $\rho = \rho'$.
2. $D(\rho,\rho') = D(\rho',\rho)$.
3. $D(\rho,\rho') + D(\rho',\rho'') \ge D(\rho,\rho'')$.
4. $D(\sum_i p_i\rho_i, \sigma) \le \sum_i p_i D(\rho_i,\sigma)$.

Fidelity

$$F(\rho,\rho') := \text{tr}\left[\sqrt{\sqrt{\rho}\rho'\sqrt{\rho}}\right]. \tag{3.80}$$

If $\rho = |\psi\rangle\langle\psi|$ and $\rho' = |\psi'\rangle\langle\psi'|$, then $F(|\psi\rangle\langle\psi|, |\psi'\rangle\langle\psi'|) = \sqrt{\langle\Psi_1|\rho_2|\Psi_1\rangle}$.
Relation to trace distance: $1 - F \le D \le \sqrt{1-F^2}$.

Min-entropy

Unconditional : $\text{H}_{\min}(X) = \text{H}_{\min}(\rho_X) = -\log \max_x p_x$.
Conditional : For a cq-state ρ_{XE}, $\text{H}_{\min}(X|E) := -\log P_{\text{guess}}(X|E)$, where

$$P_{\text{guess}}(X|E) := \max_{\{M_x\}_x} \sum_x p_x\,\text{tr}\left[M_x\rho_x^E\right], \{M_x \ge 0 \mid \sum_x M_x = \mathbb{I}\}.$$

Properties:

1. $0 \le \text{H}_{\min}(X|E) \le \text{H}_{\min}(X) \le \log|X|$, but only for cq states! For quantum register X, $\text{H}_{\min}(X|E)$ can be negative.
2. $\text{H}_{\min}(X|E) \ge \text{H}_{\min}(X) - \log|E|$.

A secret key

A key K is secret from Eve iff it is *uniform and uncorrelated* from Eve, i.e. the joint state ρ_{KE} is of the form

$$\rho_{KE} = \frac{\mathbb{I}_K}{d_K} \otimes \rho_E. \tag{3.81}$$

Bibliography

[FV99] Christopher A Fuchs and Jeroen Van De Graaf. “Cryptographic distinguishability measures for quantum-mechanical states”. In: *IEEE Transactions on Information Theory* 45.4 (1999), pages 1216–1227 (cited on page 5).

[Hel76] Carl W. Helstrom. *Quantum detection and estimation theory / Carl W. Helstrom*. English. Academic Press New York, 1976, ix, 309 p. : ISBN: 0123400503 (cited on page 3).

[KRS09] Robert Konig, Renato Renner, and Christian Schaffner. “The operational meaning of min-and max-entropy”. In: *IEEE Transactions on Information theory* 55.9 (2009), pages 4337–4347 (cited on page 9).

[Ren08] Renato Renner. “Security of quantum key distribution”. In: *International Journal of Quantum Information* 6.01 (2008), pages 1–127 (cited on page 7).

[Tom+13] M. Tomamichel et al. “A Monogamy-of-Entanglement Game With Applications to Device-Independent Quantum Cryptography”. In: *New Journal of Physics* 15 (2013). EUROCRYPT 2013, arXiv:1210.4359, page 103002 (cited on pages 16, 17).

Lecture Notes

Quantum Cryptography Week 4:

From imperfect information to (near) perfect security

Contents

This week we discuss *privacy amplification*. This task is an essential component of many cryptographic protocols; in particular it forms the final step in the quantum key distribution protocols we'll see in the coming weeks. Moreover, we'll see that privacy amplification can be achieved using a beautiful family of objects from theoretical computer science called *randomness extractors* — themselves well worth studying in their own right!

4.1 Privacy amplification

Let's start by introducing the task of privacy amplification. Imagine (as usual!) that Alice and Bob want to use cryptography to exchange messages securely. For this they have access to a classic public communication channel: they can send each other any messages they like, *but* the channel is public: the malicious eavesdropper Eve may be listening in on the whole communication. Our only cryptographic assumption on the channel is that it is *authenticated*, meaning that when Alice (or Bob) receives a message she has the guarantee that it came directly from Bob (or Alice). (We will return to the topic of authentication in Week 6; for the time being think of it as a convenient assumption that will usually be met in practice. We will also assume the channel is noiseless, which in practice is easily ensured by a proper use of error-correcting codes.)

Alice and Bob would like to use symmetric-key cryptography: they know (as you do!) that the one-time pad is unconditionally secure, so the only thing they need is to come up with a shared secret key. Moreover, Alice and Bob being old-time friends, they already have a lot of shared secrets, such as the flavor of the first ice-cream cone they shared. By putting all these secrets together and translating them in a string of bits, they're pretty confident they can come up with some value, call it $x \in \{0,1\}^n$, that's fairly secret...but only "fairly" so. Unfortunately they're not fully confident about which parts of x can be considered a secret, and which may have leaked. Alice might have told her best friend Mary about the ice-cream. She definitely wouldn't have told Mary about her (embarrassing) all-time favorite cheeky cartoon, but then her little brother John might now about this...Is there a way for Alice and Bob to somehow "boil down" the secrecy that x contains, throwing away some of the bits but without knowing a priori which are secure and which may potentially have been leaked?

Answer: yes! This is precisely what privacy amplification will do for them. To describe the task more precisely, consider the following scenario. Two mutually trusting parties, Alice and Bob, each holds a copy of the same string of bits x, which we'll call a "weak secret". This secret is taken from a certain distribution p_x, which we can represent through a random variable X; later on we'll call X the "source". The distribution of X itself is not known, but the sample x is available to both parties. An eavesdropper has side information E that may be correlated to X; for example E could be the first bit of X, the parity of X, or an arbitrary quantum state ρ_x^E. Given this setup, the goal for Alice and Bob is to each produce the same string z, which could be shorter than x but must be such that the distribution of z (represented via a random variable Z) is (close to) uniform, even from the point of view of the eavesdropper.

To summarize using symbols, privacy amplification is the transformation:

$$\rho_{XE} \qquad \xmapsto{\mathrm{PA}_X \otimes \mathbb{I}_E} \qquad \rho_{ZE} \approx_\varepsilon \frac{\mathbb{I}_Z}{|Z|} \otimes \rho_E. \tag{4.1}$$

Of course this will only be possible under some assumption on X: for example if $X = E$ always there is not much we can do. Given what we've already learned, it's natural to measure the "potential for privacy amplification" of a source X through the min-entropy (equivalently, the guessing probability) $\mathrm{H}_{\min}(X|E)$, as this is a measure of "uncertainty" the eavesdropper has about X. But we're getting ahead of ourselves. First let's see how to perform a simpler but closely related task, *randomness extraction*. Then we'll see how to use this to achieve privacy amplification.

Before diving in, consider the following warm-up exercise:

Exercise 4.1.1 Suppose that $X \in \{0,1\}^3$ is uniformly distributed, and $E = X_1 \oplus X_2 \in \{0,1\}$. Give a protocol for privacy amplification that outputs two secure bits (without any communication). What if $E = (X_1 \oplus X_2, X_2 \oplus X_3) \in \{0,1\}^2$, can you still do it? If not, give a protocol extracting just one bit.

Suppose the eavesdropper is allowed to keep any two of the bits of X as side information. Give a protocol for Alice and Bob to produce a Z which contains a single bit that is always uniformly random, irrespective of which two bits of X are stored by the eavesdropper. How about an R that contains two bits — can they do it? ∎

4.2 Randomness extractors

In the task of randomness extraction there is a single party, Alice, who has access to an n-bit string x with distribution X. We call X the *source*. X is unknown, and it may be correlated to an additional system E over which Alice has no control. For example, E could contain some information about the way in which the source was generated, or some information that an adversary has gathered during the course of an earlier protocol involving the use of X. The only promise that is given to Alice is a lower bound on the min-entropy, $\mathrm{H}_{\min}(X|E) \geq k$. Alice's goal is to produce a new string Z that is close to uniform and uncorrelated with E. (As you can see, this problem is very similar to privacy amplification, but without the added complication of Alice having to coordinate with Bob!)

Now, of course Alice could dump X and create her own uniformly random Z, say by measuring a $|0\rangle$ qubit in the Hadamard basis. To make the problem interesting we won't allow any quantum resources to Alice. She also doesn't have that much freely accessible randomness — maybe she can get some, but it will be slow and costly. So Alice's goal is to leverage what she has to the best she can: she wants to *extract* randomness from X, not import it from some magical elsewhere!

4.2.1 Randomness sources

Let's see some concrete examples of sources X, and how it is possible to extract uniform bits from them.

I.I.D. sources

The simplest case of a randomness source is the i.i.d. source, where the term i.i.d. stands for *independent and identically distributed*. A (classical) i.i.d. source $X \in \{0,1\}^n$ has a distribution $\{p_x\}$ which has a product form: there is a distribution $\{p_0, p_1\}$ on a single bit such that for all $(x_1, \ldots, x_n) \in \{0,1\}^n$,

$$\Pr[X = (x_1, \ldots, x_n)] = \Pr[X_1 = x_1] \cdots \Pr[X_n = x_n] = p_{x_1} \cdots p_{x_n}.$$

Such sources are sometimes called *von Neumann* sources, since they were already considered by von Neumann. If you are curious about the history of randomness extraction, go look up the von Neumann extractor online!

Can we extract uniformly random bits from an i.i.d. source? As a warmup, let's consider how we could obtain a nearly uniform bit from a source such that each bit X_i is 0 with probability $p_0 = 1/4$ and 1 with probability $p_1 = 3/4$. Suppose we let $Z = X_1 \oplus X_2 \oplus \ldots \oplus X_n \in \{0,1\}$ be the parity of all n bits of X. Our goal is to show that $\Pr[Z = 0] \approx 1/2 \pm \varepsilon$ for reasonably small ε.

- Let's first consider $n = 2$. How well does our strategy work? We can compute

$$\begin{aligned}\Pr[Z = 0] &= \Pr[X_1 = 0 \wedge X_2 = 0] + \Pr[X_1 = 1 \wedge X_2 = 1] \\ &= p_0^2 + p_1^2 = 1/16 + 9/16 = 0.625,\end{aligned}$$

  and using a similar calculation we find $\Pr[Z = 1] = 0.375$. Not quite uniform, but closer than what we started with!

- By doing the calculation for increasingly large values of n you will see that the trace distance ε of Z from a uniformly distributed random variable gets smaller and smaller. At what rate? Give a bound on ε as a function of n. Do you find our procedure efficient?

Independent bit sources

A slightly broader class of sources are *independent bit* sources. As their name suggests such sources are characterized by the condition that each bit is chosen independently; however the distribution could be different for different bits. Clearly, any i.i.d. source is also an independent bit source, but the converse does not hold.

Exercise 4.2.1 Show that there exists an independent 2-bit source X such that $\Pr[X = (0,0)] = \Pr[X = (1,1)] = 3/16$, but there is no i.i.d. source satisfying the same condition. ■

It turns out that taking the parity of all the bits in the string generated by an independent bit source still results in a bit that is increasingly close to uniform as $n \to \infty$, provided each bit from the source is not fully biased to start with: $0 < \Pr[X_j = 0] < 1$ for all j.

Exercise 4.2.2 Let X be an independent n-bit source such that $\delta < \Pr[X_j = 0] < 1 - \delta$ for some $\delta > 0$ and all $j \in \{1, \ldots, n\}$. Give an upper bound on the distance from uniform of the parity of the bits of x, as a function of the number of bits n of X and δ. ■

Bit-fixing sources

Bit-fixing sources are a special case of independent sources where each bit of X can be of one of two kinds only: either the bit is completely fixed, or it is uniformly random. For example, the three-bit source X such that $\Pr[X = (1,0,0)] = \Pr[X = (1,1,0)] = 1/2$, with all other probabilities being 0, is a bit-fixing source: the first bit is fixed to 1, the second is uniformly random, and the third is fixed to 0.

You can verify for yourselves that, just as for the previous two types of sources we considered, taking the parity of all bits from a bit-fixing source gives a uniformly random bit. This time, we do even better: as long as at least one of the bits from the source is not fixed, the parity is (exactly) uniformly random.

General sources

The randomness sources we just discussed all have something in common: they produce a string in which each bit is chosen *independently*. What if we relax this condition?

Consider a tricky example, called an *adversarial* bit-fixing source: this is the same as a bit-fixing source, except the value taken by the fixed bits can depend on the previous bits. For example, the three-bit source X such that $\Pr[X = (1,0,0)] = \Pr[X = (1,1,1)] = 1/2$, with all other probabilities being 0, is an adversarial bit-fixing source: the first bit is fixed to 1, the second is uniformly random, and the third is fixed to, either 0 if the second was a 0, or 1 if the second was a 1. To see that this kind of source can be much more tricky, first check that our earlier choice of Z as the parity of all the bits of X no longer works on the example. However, the parity of the first two, or the first and last, bits does work on that example. Nevertheless, show that for any fixed choice of a subset of bits, there exists an adversarial bit-fixing source such that only one bit is fixed, but nevertheless the parity of the bits in the chosen subset is a constant — arbitrarily far from uniform!

As you can imagine there is a whole jungle of possible kinds of sources. How do we classify them? For the purposes of extracting randomness, we aim to measure the inherent uncertainty of the source, or in other words its *entropy*. It turns out that the min-entropy provides just the "right" measure of extractable randomness, in a precise way that we'll soon see.

Definition 4.2.1 A random variable X is a k-source if $\mathrm{H}_{\min}(X) \geq k$.

Before we move on, we should realize there is something crucial missing from this definition. Remember we're going to apply the idea of randomness extraction to a cryptographic task, privacy amplification. But we forgot to account for the eavesdropper! The process of randomness extraction is not going to happen in a void: we ought to take into account the possibility for an additional system E that may be correlated with X. Call E the *side information*. X is a classical string of bits, but E may be quantum. How do we model this? The proper way to do it is to introduce a cq state ρ_{XE}, which in general takes the form

$$\rho_{XE} = \sum_x |x\rangle\langle x|_X \otimes \rho_x^E,$$

where each ρ_x^E is positive semidefinite and $\mathrm{Tr}(\rho_{XE}) = \sum_x \mathrm{Tr}(\rho_x^E) = 1$. Using side information gives us a convenient way to model any source X as the result of an initially uniform string about which the adversary has gained partial information. For instance, you can think of a bit-fixing source as a uniform source correlated with a system E which contains some of the bits of x.

Exercise 4.2.3 Let X be an independent source, where the i-th bit X_i has distribution $\{p_i, 1-p_i\}$. Show that there exists a pair of correlated random variables (Y,Z) on $\{0,1\}^n \times \{0,1\}^n$ such that Y is uniformly distributed in $\{0,1\}^n$ but for any $z \in \{0,1\}^n$ the random variable $V = Y_{|Z=z}$ is such that V_i has the same distribution as X_i if $z_i = 0$, and as $1 - X_i$ if $z_i = 1$. ■

Let's update our definition:

Definition 4.2.2 A cq state ρ_{XE} is called a k-source if $\mathrm{H}_{\min}(X|E) \geq k$.

Can we construct extraction procedures that produce uniformly random bits from any k-source, without knowing anything else about the source?

4.2.2 Strong seeded extractors

In all examples we've seen so far we applied a fixed function, call it Ext, to the source X; for example we considered $\mathrm{Ext}(X) = X_1 \oplus \cdots \oplus X_n$. Such a function is known as a deterministic extractor, meaning that is it just one fixed function $Z = \mathrm{Ext}(X)$ that does not introduce any randomness beside what is already present in X.

Ideally we'd like to show that it is possible to extract randomness from any k-source using such a deterministic function. Unfortunately this is not possible: there is no fixed deterministic procedure that can be used to extract even a *single* bit of randomness from every possible k-source, even when k is almost maximal, $k = n-1$! This is a bit disappointing, but let's understand why.

Lemma 1 For any function $\mathrm{Ext} : \{0,1\}^n \to \{0,1\}$ there exists an $(n-1)$-source X such that $\mathrm{Ext}(X)$ is constant.

Proof. Let $b \in \{0,1\}$ be such that $|S_b| \geq 2^n/2 = 2^{n-1}$ with $S_b = \{x \mid \mathrm{Ext}(x) = b\}$. Note that there must exist such a b. Choose a subset $S' \subseteq S_b$ such that $|S'| = 2^{n-1}$. Define X by the following distribution:

$$p_x = \begin{cases} 1/2^{n-1} & \text{if } x \in S' \,, \\ 0 & \text{otherwise} \,. \end{cases} \tag{4.2}$$

Clearly, $\mathrm{H}_{\min}(X) = n-1$, but $\mathrm{Ext}(X) = b$ is a constant! ■

Have we reached the end of the road — are we stuck to designing special-purpose functions which only work for this or that special kind of source, as we did with independent sources? Luckily

there is a way out, but we're going to need an additional resource: a little extra randomness. This extra randomness will be called the *seed* of the extractor; think of it as a second input $Y \in \{0,1\}^d$ to which Alice has access and is promised to be uniformly random and independent from X and E. This gives us the notion of a *seeded extractor*:

Definition 4.2.3 A (k,ε)-*weak seeded randomness extractor* is a function $\mathrm{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ such that for any k-source ρ_{XE},

$$D\Big(\rho_{\mathrm{Ext}(X,Y)E}, \frac{\mathbb{I}}{2^m} \otimes \rho_E\Big) \leq \varepsilon \,, \tag{4.3}$$

where $Y \sim U_d$ is uniformly distributed and independent from X and E, and

$$\rho_{\mathrm{Ext}(X,Y)E} = \sum_z |z\rangle\langle z|_Z \otimes \rho_z^E \qquad \text{with} \qquad \rho_z^E = 2^{-d} \sum_y \sum_{x:\mathrm{Ext}(x,y)=z} \rho_x^E .$$

If the seed is perfectly uniform, why don't we just return it as our output: define $\mathrm{Ext}(X,Y) = Y$? Well, this satisfies the definition. So maybe there is something wrong with the definition? Remember that our goal is to extract randomness from X, and that additional uniform randomness should not be considered free. So we want to keep Y as small as possible, even though X, and k, could be very large, in which case we'd like to maintain a long output (large m) with only a small help from the seed (small d).

A better answer considers our ultimate goal of privacy amplification. Remember that in that setting Alice and Bob share a weak secret X, and they want to produce a uniformly random secret R. Our solution of an extractor outputting its seed would be similar to asking Alice and Bob to throw away their initial secret X and share a fresh random string Y. But they only have access to a public communication channel, how would they agree on the same Y without the eavesdropper learning it as well?

This motivates a stronger definition of extractor, which is the one we'll use from now on:

Definition 4.2.4 A (k,ε)-*strong seeded randomness extractor* is a function $\mathrm{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ such that for any k-source ρ_{XE},

$$D\big(\rho_{\mathrm{Ext}(X,Y)YE}, \frac{\mathbb{I}}{2^m} \otimes \rho_{YE}\big) \leq \varepsilon \,, \tag{4.4}$$

where $Y \sim U_d$ is uniformly distributed and independent from X and E.

Before we start trying to construct strong extractors, let's consider the notion of k-source a little more closely. Why do we think that the min-entropy provides the right way to quantify the amount of randomness that can be extracted from a given source?

Let's first argue informally that the min-entropy is an upper bound on the amount of extractable randomness: there is no strong extractor that has output length more than $\mathrm{H}_{\min}(X|E)$. To see why this is the case, recall that $\mathrm{H}_{\min}(X|E) = -\log P_{\mathrm{guess}}(X|E)$. Suppose now that we apply some function f to X. How hard is it to guess $f(X)$ given E, i.e., what's $P_{\mathrm{guess}}(f(X)|E)$? Clearly, since one way to guess $f(X)$ is to guess X, and then apply f to our guess, we have $P_{\mathrm{guess}}(f(X)|E) \geq P_{\mathrm{guess}}(X|E)$. However, this is equivalent to

$$\mathrm{H}_{\min}(f(X)|E) \leq \mathrm{H}_{\min}(X|E) \,. \tag{4.5}$$

This means that also the output of the extractor, which for fixed seed y is obtained as a function $f(X) = \mathrm{Ext}(X,y)$, must have min-entropy at most $\mathrm{H}_{\min}(X|E)$, which implies that the output $\mathrm{Ext}(X,Y)$, conditioned on $Y = y$, can be uniform on at most $\mathrm{H}_{\min}(X|E)$ bits!

Now, how about a converse: does there exist a strong extractor that can extract approximately

$\mathrm{H}_{\min}(X|E)$ bits from *any* k-source ρ_{XE}? The answer turns out to be yes, and we're going to see how this can be done in the next section.

4.3 An extractor based on hashing

Much research has gone into constructing randomness extractors, and they have found many applications throughout computer science and mathematics. The quality of an extractor is measured by the parameters it achieves, and different applications require different trade-offs. The main targets consist in extracting as much randomness as possible (large m) using the smallest possible seed (small d) and with the best possible error (small ε), all from arbitrary sources with min-entropy (at least) k.

By using a probabilistic argument (select a function Ext at random from all possible functions, and fix it to be the extractor), for any given input length k and min-entropy k the best trade-offs that can be achieved are seed length $d = \log(n-k) + 2\log(1/\varepsilon) + O(1)$ and output length $m = k + d - 2\log(1/\varepsilon) + O(1)$ [**radhakrishnan2000bounds**]. Moreover, there are efficient constructions known that achieve essentially both parameters simultaneously! Rather than aiming for optimal, but often intricate, constructions, here we will focus on a simple construction which nevertheless achieves very good parameters for the application we have in mind (privacy amplification!).

Going back to the intuition we developed on the examples, we saw that taking the parity of a random subset of the bits of the source often (but not always) provides a good way to extract a bit of randomness. In this case we can think of the seed of the extractor as specifying the subset of bits whose parity is taken. We could repeat this procedure to extract even more bits, each chosen as the parity of a different random subset. It is a good exercise to show that this procedure works, but it has one major drawback: it is excessively costly in terms of seed length, requiring an investment of approximately n bits of randomness (to specify the subset of bits whose parity is taken) for each new bit produced!

Let's see how we can do better. For this we'll have to make a little detour and learn about certain families of hash functions.

4.3.1 Two-universal families of hash functions

Informally, a hash function is a function that maps long strings to shorter strings, with the property that the output of the hash functions tends to be "well-distributed". What this means depends on the application we have in mind for the hash function — indeed, the term "hash function" can be interpreted in many different ways, with the only standard requirement, as its name indicates, being that a hash function should not increase the length of its input! An additional reasonable requirement, which formalizes the "well-distributed" aspect of the output of a hash function, is the following:

Definition 4.3.1 — 1-universal family. A family of hash functions $\mathscr{F} = \{f : \{0,1\}^n \to \{0,1\}^m\}$, where $m \leq n$, is called 1-*universal* if for every string $x \in \{0,1\}^n$ and $z \in \{0,1\}^m$ we have

$$\Pr_{f \in \mathscr{F}}[f(x) = z] = \frac{1}{2^m} . \tag{4.6}$$

It is worth reading this definition carefully: in (4.6) both x and z are fixed, and the probability is taken over a uniformly random function from the family. The condition is equivalent to saying that for any fixed x, the random variable $F(x)$, where F is uniformly distributed over all f in $\mathscr{F}$, in uniformly distributed in $\{0,1\}^m$. Let's see an example of a 1-universal family of hash functions.

Exercise 4.3.1 For any $y \in \{0,1\}^n$ let $f_y : \{0,1\}^n \to \{0,1\}^n$ be defined by $f_y(x) = x \oplus y$, where the parity is taken bitwise. Show that the family of functions $\mathscr{F} = \{f_y, y \in \{0,1\}^n\}$ is 1-universal. ■

You may want to convince yourself that a family of 1-universal hash functions is already sufficient to construct a *weak* seeded extractor: use the seed to select a random function from the family, and output the value of the function evaluated at the source. The property of 1-universality ensures that the output will be uniformly distributed, even if the input is fixed. However, recall our earlier criticism: in this case it is apparent that we are "cheating", and that all the randomness is coming from the seed. Indeed, it turns out that the property of 1-universality is not sufficient to obtain a *strong* seeded extractor. We'll need the following stronger property, first introduced by Carter and Wegman:

Definition 4.3.2 — 2-universal family. A family of hash functions $\mathscr{F} = \{f : \{0,1\}^n \to \{0,1\}^m\}$ is called 2-*universal* if for every two strings $x, x' \in \{0,1\}^n$ with $x \neq x'$, and any two $z, z' \in \{0,1\}^m$, we have

$$\Pr_{f \in \mathscr{F}}[f(x) = z \wedge f(x') = z'] = \frac{1}{2^{2m}} . \tag{4.7}$$

Condition (4.7) in the definition would be satisfied if $f(x)$ and $f(x')$ were *jointly* chosen uniformly and independently at random in $\{0,1\}^m$. This is a stronger condition than (4.6): we now require that the pair of random variables $(F(x), F(x'))$, for F uniformly distributed over $f \in \mathscr{F}$, are jointly uniform (as an exercise, verify that the family of hash functions from Exercise 4.3.1 is *not* 2-universal).

You can check that for any $m \leq n$ the set of all possible functions $f : \{0,1\}^n \to \{0,1\}^m$ is 2-universal. But it is too big a set: it has size $|\mathscr{F}| = 2^{m2^n}$, so that selecting a function at random from the set would require a seed length $d = m2^n$! Let's see a much more efficient construction.

Let $q = 2^n$ and $\mathbb{F}_q$ the finite field with 2^n elements. (If you have never seen this field before, the details of its construction will not be matter to us, but you may still want to check it out online! The multiplication rule is *not* the same as multiplication over the integers, $\mod 2^n$.) For any $(a,b) \in \mathbb{F}_q^2$ let

$$f_{a,b} : \mathbb{F}_q \to \mathbb{F}_q, \qquad f_{a,b}(x) = ax + b,$$

where addition and multiplication are done in $\mathbb{F}_q$. Then $\mathscr{F} = \{f_{a,b}, (a,b) \in \mathbb{F}_q^2\}$ is a 2-universal family of only $q^2 = 2^{2n}$ hash functions. To show this we need to verify that equation (4.7) from the definition holds. So let's fix distinct $x \neq x' \in \mathbb{F}_q$ and two $z, z' \in \mathbb{F}_q$. What is the probability, over a uniformly random choice of (a,b), that $f_{a,b}(x) = z$ and $f_{a,b}(x') = z'$? The two conditions are equivalent to $ax + b = z$ and (taking the difference) $a(x' - x) = z' - z$, thus $a = (z' - z)/(x' - x)$, where the condition $x \neq x'$ and the fact that $\mathbb{F}_q$ is a field allows us to perform the division. This equation determines a unique possible value for a. Moreover, once a is fixed there is a unique possible value for b: $b = z - ax$ (this shouldn't be a surprise, since we started with two linear equations and two unknowns). Out of 2^{2m} possibilities, we end up with a single one: $\Pr_{a,b}[f_{a,b}(x) = z \wedge f_{a,b}(x') = z'] = 2^{-2m}$, as desired.

One last technicality: recall that our goal was to construct a 2-universal family of functions $f : \{0,1\}^n \to \{0,1\}^m$, for arbitrary n and $m \leq n$, whereas what we managed to construct so far are functions from $\mathbb{F}_q \to \mathbb{F}_q$. Since $|\mathbb{F}_q| = q = 2^n$ the domain of f can be identified with $\{0,1\}^n$ in an arbitrary way. The range of f may be bigger than $\{0,1\}^m$, but there is a simple solution: throw away the last $(n-m)$ bits of $f(x)$! I'll let you verify that this works, i.e. it preserves the property of 2-universality.

4.3.2 The 2-universal extractor

Equipped with an arbitrary family of 2-universal hash functions, we define an extractor as follows.

Definition 4.3.3 — 2-universal extractor. Let $\mathscr{F} = \{f_y : \{0,1\}^n \to \{0,1\}^m, y \in \{0,1\}^d\}$ be a 2-universal family of hash functions such that $|\mathscr{F}| = 2^d$. The associated 2-universal extractor is

$$\mathrm{Ext}_{\mathscr{F}} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m, \qquad \mathrm{Ext}_{\mathscr{F}}(x,y) = f_y(x).$$

One way to think of $\mathrm{Ext}_{\mathscr{F}}$ is as using its seed y to select a function from the family $\mathscr{F}$ uniformly at random, and then returning the output of the function when evaluated on the source X.

How good is this extractor? The key result required to analyze it is known as the *leftover hash lemma*. It was first proven by Impagliazzo, Levin, and Luby for the case when there is no side information E, and later extended to the case of quantum E by Renner. Here is a statement of the lemma when there is no side information.

Theorem 4.3.1 — Leftover hash lemma. Let n and $k \leq n$ be arbitrary integers, $\varepsilon > 0$, $m = k - 2\log(1/\varepsilon)$, and $\mathscr{F} = \{f : \{0,1\}^n \to \{0,1\}^m\}$ a 2-universal family of hash functions. Then the 2-universal extractor $\mathrm{Ext}_{\mathscr{F}}$ is a (k,ε)-strong seeded randomness extractor.

In the previous section we saw how to construct a 2-universal family with 2^{2n} functions, meaning that the seed length of the two-universal extractor is $2n$. This is much longer than the optimal length $d \approx O(\log(n/\varepsilon))$, and it can be a drawback in some applications for which the randomness required to produce the seed is particularly costly. However, for our application to privacy amplification, and especially later to quantum key distribution, it is not a significant limitation. Much more important for us is the dependence of the output length on the initial min-entropy, which will ultimately govern the length of key that we are able to produce. In this respect the two-universal construction is essentially optimal, a good reason to use it!

4.3.3 Analysis with no side information

We first prove the leftover hash lemma in the case when there is no side information, stated in Theorem 4.3.1. This will be a good warm-up for the general case, which will follow the same structure.

The proof proceeds in two steps. In the first step we reduce our ultimate goal, bounding the error of the extractor, i.e. the trace distance between the extractor's output and the uniform distribution, to bounding a different quantity called the *collision probability*. In the second step we show that the collision probability is sufficiently small to imply the desired bound on the error of the extractor.

(i) From trace distance to collision probability.

Our goal is to bound $D(\rho_{\mathrm{Ext}(X,Y)Y}, 2^{-(m+d)}\mathbb{I})$, where X has min-entropy at least k and Y is uniformly distributed over d-bit strings. The joint distribution of $(Z = \mathrm{Ext}(X,Y), Y)$ is given by

$$p_{zy} = \Pr[(\mathrm{Ext}(X,Y),Y) = (z,y)] = 2^{-d} \sum_{x : f_y(x) = z} p_x. \tag{4.8}$$

Using the definition of the trace distance, we get

$$\begin{aligned} D(\rho_{\mathrm{Ext}(X,Y)Y}, 2^{-(d+m)}\mathbb{I}) &= \frac{1}{2}\sum_{z,y} \left| 2^{-d} \sum_{x : f_y(x) = z} p_x - 2^{-d-m} \right| \\ &\leq 2^{\frac{m}{2}-1} \left(2^{-d} \sum_{z,y} \left| \sum_{x : f_y(x) = z} p_x - 2^{-m} \right|^2 \right)^{1/2} \\ &= 2^{\frac{m}{2}-1} \left(2^{d} \sum_{z,y} p_{zy}^2 - 2^{-m} \right)^{1/2}, \end{aligned}$$

where for the second line we applied the Cauchy-Schwarz inequality. This completes our first step. The quantity $CP(ZY) = \sum_{z,y} p_{zy}^2$ is called the collision probability of (Z,Y), and we turn to bounding it next.

(ii) A bound on the collision probability.
Using the definition (4.8) and expanding the square,

$$\begin{aligned}\sum_{z,y} p_{zy}^2 &= 2^{-2d} \sum_{y,z} \sum_{\substack{x,x':\\ f_y(x)=f_y(x')=z}} p_x p_{x'} \\ &= 2^{-2d} \sum_{y,z} \Big(\sum_{\substack{x\neq x':\\ f_y(x)=f_y(x')=z}} p_x p_{x'} + \sum_{x:\, f_y(x)=z} p_x^2 \Big) \\ &= 2^{-(d+m)} \sum_{x\neq x'} p_x p_{x'} + 2^{-d} \sum_x p_x^2 \\ &\leq 2^{-(d+m)} + 2^{-(d+k)}.\end{aligned}$$

Here the crucial step is in bounding the summation over $x \neq x'$ when going from the second to the third line: we are using the property of 2-universality to argue that for any $x \neq x'$ there is a fraction exactly 2^{-m} of all f_y that map both x and x' to the same value. To bound the second term in going from the second-last to last lines we used $\sum_x p_x^2 \leq \max_x p_x = 2^{-\mathrm{H}_{\min}(X)}$ and the assumption $\mathrm{H}_{\min}(X) \geq k$.

Plugging this back into the bound on the trace distance from (i) we obtain

$$D(\rho_{\mathrm{Ext}(X,Y)Y}, 2^{-(d+m)}\mathbb{I}) \leq 2^{\frac{m-k}{2}-1},$$

proving the lemma.

4.3.4 The pretty good measurement and quantum side information

We would like to extend the proof in the previous section to the case where the source X is correlated with some quantum side information E, that is, $\rho_{XE} = \sum_x |x\rangle\langle x| \otimes \rho_x^E$ is an arbitrary cq state such that $\mathrm{H}_{\min}(X|E) \geq k$. Before diving into this, let's make a small detour by considering the related problem of optimally distinguishing between a set of quantum states.

The pretty-good measurement
Let $\rho_{XE} = \sum_x |x\rangle\langle x| \otimes \rho_x^E$ be a cq state. What is the optimal probability with which Eve, holding the quantum system E, can successfully guess x? We've seen this problem already: the answer is captured by the guessing probability,

$$P_{\mathrm{guess}}(X|E)_\rho = \max_{\{M_x\}} \sum_x \mathrm{Tr}(M_x \rho_x^E), \tag{4.9}$$

where the maximum is taken over all POVM $\{M_x\}$ on E. But what is the best POVM? If $x \in \{0,1\}$ takes only two values you've already seen the answer: in this case we can write

$$\begin{aligned}\mathrm{Tr}(M_0\rho_0^E) + \mathrm{Tr}(M_1\rho_1^E) &= \mathrm{Tr}\left(\frac{M_0+M_1}{2} \cdot \left(\rho_0^E + \rho_1^E\right)\right) + \mathrm{Tr}\left(\frac{M_0-M_1}{2} \cdot \left(\rho_0^E - \rho_1^E\right)\right) \\ &\leq \frac{1}{2} + \frac{1}{2} D(\rho_0^E, \rho_1^E),\end{aligned}$$

and moreover the last inequality is an equality if M_0 and M_1 are the projectors on the positive and negative eigenspaces of the Hermitian matrix $\rho_0^E - \rho_1^E$ respectively.

When $|X| > 2$ unfortunately the situation is a bit more murky. The problem of finding the optimal measurement can be solved efficiently with a computer by expressing the optimization problem (4.9) as a *semidefinite program*, a generalization of linear programs for which there are efficient algorithms. But what we'd really like is a nice, clean mathematical expression for what the optimal measurement is, so that we can work with it in our proofs! No such simple closed form is known. However, what we can do is find a simple measurement that always achieves *close* to the optimum: the *pretty-good measurement*.

So what is this "pretty-good" measurement? To get some intuition first consider the case where the states ρ_x^E are perfectly distinguishable; for example $\rho_x^E = p_x|x\rangle\langle x|$ is simply a classical copy of X. Then it is clear what we should do: measure in the computational basis, and recover x! Observe that in this case the POVM elements M_x are directly proportional to ρ_x: we can think of the states as "pointing" in some direction correlated with x, and it is natural to make a measurement along that direction.

Can we generalize this idea? Let's try defining $M_x = \rho_x^E$. This is positive semidefinite, so it satisfies the first condition for a POVM. However, $\sum_x M_x = \sum_x \rho_x^E = \rho^E$ is not necessarily the identity, as required by the second condition. The solution? Normalize!

Definition 4.3.4 Given a collection of positive semidefinite matrices $\{\rho_x\}$, the *pretty-good measurement* (PGM) associated to the collection is the POVM with elements

$$M_x = \rho^{-1/2}\rho_x\rho^{-1/2},$$

where $\rho = \sum_x \rho_x$ and the inverse is the Moore-Penrose pseudo-inverse, i.e. we use the convention $0^{-1} = 0$.

Note how we dealt with division by zero in the definition. Defining division by zero may seem odd, but this convention makes sense in the context of linear operators. If ρ is orthogonal to some subspace, i.e. it is an eigenspace of eigenvalue 0, then the pseudo-inverse ρ^{-1} should also be orthogonal to that subspace. A useful property of this convention is that it makes it so that if P is an orthogonal projection and $P\rho P$ is invertible, then $(P\rho P)^{-1} = P\rho^{-1}P$.

How well does the pretty-good measurement compare to the optimal guessing measurement? Let $\{N_x\}$ be an optimal guessing POVM for Eve. Then by definition

$$\begin{aligned}
P_{\text{guess}}(X|E) &= \sum_x \operatorname{Tr}\left(N_x\rho_x^E\right) \\
&= \sum_x \operatorname{Tr}\left((\rho^{1/4}N_x\rho^{1/4})(\rho^{-1/4}\rho_x^E\rho^{-1/4})\right) \\
&\leq \Big(\sum_x \operatorname{Tr}\left(\rho^{1/2}N_x\rho^{1/2}N_x\right)\Big)^{1/2}\Big(\sum_x \operatorname{Tr}\left(\rho^{-1/2}\rho_x^E\rho^{-1/2}\rho_x^E\right)\Big)^{1/2} \\
&\leq \left(\text{PGM}(X|E)\right)^{1/2},
\end{aligned}$$

where

$$\text{PGM}(X|E) = \sum_x \operatorname{Tr}(M_x\rho_x^E) = \sum_x \operatorname{Tr}\left(\rho^{-1/2}\rho_x\rho^{-1/2}\rho_x\right) \tag{4.10}$$

is the success probability of the PGM in the guessing task. The second and third lines are the most important here. To go from the first to the second line we inserted factors $\rho^{1/4}$ and $\rho^{-1/4}$ that cancel each other out (using cyclicity of the trace), but are important for normalization. To go from the second to the third line we used the Cauchy-Schwarz inequality twice: first, for each x we apply a matrix version of the inequality,

$$\left|\operatorname{Tr}(AB)\right| \leq \left(\operatorname{Tr}(AA^\dagger)\right)^{1/2}\left(\operatorname{Tr}(BB^\dagger)\right)^{1/2}, \tag{4.11}$$

with $A = \rho^{1/4} N_x \rho^{1/4}$ and $B = \rho^{-1/4} \rho_x^E \rho^{-1/4}$; and second, we apply the usual version

$$\Big|\sum_x a_x b_x\Big| \leq \Big(\sum_x a_x^2\Big)^{1/2} \Big(\sum_x b_x^2\Big)^{1/2},$$

valid for any real a_x and b_x (here $a_x = \mathrm{Tr}(\rho^{1/2} N_x \rho^{1/2} N_x)$ and $b_x = \mathrm{Tr}(\rho^{-1/2} \rho_x^E \rho^{-1/2} \rho_x^E)$). Finally to get to the last line we used $\sum_x N_x = \mathbb{I}$ to bound the first term, and the definition of the pretty-good measurement for the second.

Proof of the leftover hash lemma with quantum side information

The proof follows the same structure as the proof we saw for the case with no side information, but it is slightly more involved technically. We will use the following inequality:for any positive Hermitian σ and positive semidefinite τ such that $\mathrm{Tr}(\tau) = 1$ and the support of τ contains the support of σ,

$$\mathrm{Tr}\big(|\sigma|\big) \leq \mathrm{Tr}\big((\tau^{-1/4} \sigma \tau^{-1/4})^2\big)^{1/2}. \tag{4.12}$$

To prove the inequality, observe that

$$\begin{aligned}
\mathrm{Tr}\big(|\sigma|\big) &= \mathrm{Tr}\big(\tau^{1/4} \tau^{-1/4} |\sigma| \tau^{-1/4} \tau^{1/4}\big) \\
&= \mathrm{Tr}\big(\tau^{1/4} |\tau^{-1/4} \sigma \tau^{-1/4}| \tau^{1/4}\big) \\
&= \mathrm{Tr}\big(|\tau^{-1/4} \sigma \tau^{-1/4}| \tau^{1/2}\big).
\end{aligned}$$

Here the second line is obtained by computing the trace in the eigenbasis of $\tau^{-1/4} \sigma \tau^{-1/4}$; see [**renner2008security**] for details of the calculation. To conclude the proof of (4.12), apply (4.11) with the choise $A = \tau^{-1/4} \sigma \tau^{-1/4}$ and $B = \tau^{1/2}$.

(i) From trace distance to collision probability.

Our goal is to bound $D(\rho_{\mathrm{Ext}(X,Y)YE}, 2^{-(m+d)} \mathbb{I} \otimes \rho_E)$, where Y is uniformly distributed and X is such that $\mathrm{H}_{\min}(X|E) \geq k$. We can write

$$\rho_{\mathrm{Ext}(X,Y)YE} = \sum_{z,y} |z\rangle\langle z| \otimes |y\rangle\langle y| \otimes \rho_{zy}, \qquad \text{with} \qquad \rho_{zy} = 2^{-d} \sum_{x:\, f_y(x) = z} \rho_x.$$

Note that our normalization makes is so that

$$\sum_{z,y} \mathrm{Tr}(\rho_{zy}) = 2^{-d} \sum_{x,y} \mathrm{Tr}(\rho_x) = \mathrm{Tr}(\rho) = 1.$$

Since the state $\rho_{\mathrm{Ext}(X,Y)YE}$ is a ccq state, using the definition of the trace distance we can expand

$$\begin{aligned}
D(\rho_{\mathrm{Ext}(X,Y)YE}, 2^{-(d+m)} \mathbb{I} \otimes \rho_E) &= \frac{1}{2} \sum_{z,y} \big\| \rho_{zy} - 2^{-(d+m)} \rho \big\|_1 \\
&\leq 2^{\frac{m+d}{2} - 1} \Big(2^{-(m+d)} \sum_{z,y} \mathrm{Tr}\big((\rho^{-1/4} (\rho_{zy} - 2^{-m} \rho) \rho^{-1/4})^2\big)\Big)^{1/2} \\
&= 2^{\frac{m}{2} - 1} \Big(2^d \sum_{z,y} \mathrm{Tr}\big(\rho_{zy} \rho^{-1/2} \rho_{zy} \rho^{-1/2}\big) - 2^{-m}\Big)^{1/2},
\end{aligned}$$

where for the second line we first applied (4.12) for each (y,z) with $\sigma = \rho_{zy} - 2^{-(d+m)} \rho$ and $\tau = \rho$, and then the usual Cauchy-Schwarz inequality. Do you recognize the expression in the last line? Using the notation from (4.10), we have

$$\mathrm{PGM}(Z|YE) = 2^d \sum_z \mathrm{Tr}\big(\rho_{zy} \rho^{-1/2} \rho_{zy} \rho^{-1/2}\big),$$

so the sequence of equations above show that

$$D(\rho_{\mathrm{Ext}(X,Y)YE}, 2^{-(d+m)}\mathbb{I}\otimes\rho_E) \leq 2^{\frac{m}{2}-1}\big(\mathrm{PGM}(Z|YE)-2^{-m}\big)^2.$$

We have thus managed to relate the distance from uniform to the advantage of the pretty good measurement over random guessing (that would succeed with probability 2^{-m}). We can understand this step of the proof as a reduction from arbitrary attacks of an adversary to the extractor, whose optimal success probability is expressed in the first line, to attacks of a very specific form, where the adversary, given a sample (z,y), measures its side information using the pretty-good measurement associated with the family of states $\{\rho_{zy}\}$. The square root factor on the third line expresses the fact that the pretty-good measurement is quadratically far from optimal. What is the point of losing this square root? The pretty-good measurement has a crucial advantage, that we are going to use in the second step of the proof: it has a form of "linearity", in the sense that the PGM operators associated with the family of states $\{\rho_{zy}\}$ can be obtained by summing up PGM operators associated with the states $\{\rho_x\}$. Let's see how this works in our favor.

(ii) A bound on the collision probability.
Proceeding exactly as in the case with no side information, we can calculate

$$\begin{aligned}
\mathrm{PGM}(Z|YE)-2^{-m} &= 2^{-d}\sum_{y,z}\sum_{\substack{x,x':\\ f_y(x)=f_y(x')=z}}\mathrm{Tr}(\rho_x\rho^{-1/2}\rho_{x'}\rho^{-1/2})-2^{-m}\\
&= 2^{-d}\sum_{y,z}\Bigg(\sum_{\substack{x\neq x':\\ f_y(x)=f_y(x')=z}}\mathrm{Tr}(\rho_x\rho^{-1/2}\rho_{x'}\rho^{-1/2})\\
&\qquad\qquad +\sum_{x:f_y(x)=z}\mathrm{Tr}(\rho_x\rho^{-1/2}\rho_x\rho^{-1/2})\Bigg)-2^{-m}\\
&= 2^{-m}\sum_{x\neq x'}\mathrm{Tr}(\rho_x\rho^{-1/2}\rho_{x'}\rho^{-1/2})+\sum_x\mathrm{Tr}(\rho_x\rho^{-1/2}\rho_x\rho^{-1/2})-2^{-m}\\
&\leq \mathrm{PGM}(X|E).
\end{aligned}$$

Using the 2-universal hashing property, we have managed to relate the advantage over random of the pretty good measurement in guessing Z, to the success probability of the pretty good measurement to guess X directly. But the last expression is, by assumption, at most $2^{-\mathrm{H}_{\min}(X|E)}$, since the guessing probability achieved from using the PGM cannot be the optimal one. Together with the bound proven in step (i) we finally obtain

$$D(\rho_{\mathrm{Ext}(X,Y)Y}, 2^{-(d+m)}\mathbb{I}) \leq 2^{\frac{m-k}{2}-1},$$

precisely the same bound as when there was no side information at all.

4.4 Solving privacy amplification using extractors

Back to cryptography...how do we use extractors to solve privacy amplification? By now you must have a good idea how this can be done. Let Ext be a (k,ε) strong seeded randomness extractor. Here is a simple protocol:

1. Alice and Bob share a weak secret X, which may be correlated with an eavesdropper holding quantum side information E.
2. Alice choses a random seed Y for the extractor, and computes $R_A = \mathrm{Ext}(X,Y)$. She sends Y to Bob over a public communication channel.
3. Upon receiving Y, Bob sets $R_B = \mathrm{Ext}(X,Y)$.

First note that this protocol is always correct: Alice and Bob output the same string, $R_A = R_B$. Is it secure? Remember the criterion (4.1) we introduced to define security of privacy amplification. Note also that here, at the end of the protocol, Eve has access to her original side information E, but also to any communication exchanged over the public channel: precisely the seed Y. So the condition becomes

$$X : \mathrm{H}_{\min}(X|E)_\rho \geq k \quad \overset{\text{PA}}{\longmapsto} \quad R = \mathrm{Ext}(X,Y) : \rho_{RYE} \approx_\varepsilon \frac{\mathbb{I}_R}{|R|} \otimes \rho_{YE},$$

which is precisely the requirement of a (k, ε) strong extractor! All the pieces have come into place: by instantiating the extractor with the 2-universal extractor based on the 2-universal family of hash functions from Section 4.3.1 you now have a complete construction of a secure one-way protocol for privacy amplification. This will be crucially used in our quantum key distribution protocols.

Acknowledgments

 The lecture notes are written by Nelly Ng, Thomas Vidick and Stephanie Wehner. We thank David Elkouss, Kenneth Goodenough, Jonas Helsen, Jérémy Ribeiro, and Jalex Stark for proofreading. We thank Joe Renes for spotting a typo in a previous version of the notes, and suggesting a streamlined analysis of the leftover hash lemma with quantum side information.

Lecture Notes

edX Quantum Cryptography: Week 5

Distributing Keys

Contents

This week, we're finally distributing actual keys! We will approach our objective in a series of steps, ending up with the famous BB84 quantum key distribution (QKD) protocol. Throughout, we will assume that Alice and Bob share an *classical authenticated channel (CAC)*. Whenever Alice and Bob make use of this channel below, we will say they send the information over "the CAC".

Let us now start our investigations into quantum cryptography. Imagine that we have several parties engaging in some communication protocol in order to solve a specific task. For example, the parties may just want to send information, or they may want to search a database, or engage in an auction. The goal of cryptography is to protect the honest parties from the dishonest ones during such an interaction. What does it mean to be honest or dishonest?

Definition 5.0.1 — Honest and dishonest (informal). Given any communication protocol between several parties:

- A party is call *honest* if he/she follows the protocol precisely. That is, the party will give the correct input to the protocol, execute all steps as dictated, and produce the desired output.
- A party is called *dishonest* or *malicious*, if he/she does not follow the protocol. Instead, this party can deviate from the protocol aribtrarily. A malicious party is also called an *adversary*.

When designing any cryptographic protocol, we first have to ask ourselves what kind of malicious parties we want to protect against. That is, whether there are some limits to what a malicious party can do in order to break the protocol, and how many resources he has at his disposal. A standard assumption that is generally implicit in all cryptographic protocols is that the honest party, let us call her Alice, sits in an impenetrable lab that the adversary does not have control over. In other words, Alice can perform local computations without the adversary's knowledge. Only when Alice sends any information out of her own lab along a communication channel does the adversary have any opportunity to intercept or tamper with the protocol execution. We will see later that quantum information makes it possible to weaken this demand! For the moment, however, we will assume that an honest Alice (or Bob) has full control over her lab.

5.1 Secure key distribution

The main cryptographic challenge that we will consider is the one of key distribution. Here, our protagonists, Alice and Bob want to protect their communication from the prying eyes of an eavesdropper Eve. Alice and Bob are thereby always honest, and Eve is the adversary. Alice and Bob have control over their secure labs that Eve cannot peek into. However, Eve has access to the communication channel connecting Alice and Bob.

Definition 5.1.1 — Key distribution. A *key distribution* protocol between Alice and Bob aims to achieve the following goals, given a security parameter $\varepsilon \geq 0$:

- $\varepsilon-$ correctness: Alice and Bob both agree on an m-bit key $K \in \{0,1\}^m$, except with some failure probability at most ε. That is, both Alice and Bob have K_A, K_B respectively, and $\text{Prob}(K_A \neq K_B) \leq \varepsilon$.
- $\varepsilon-$ security: Any outsider Eve is almost ignorant about the key, i.e. $\rho_{KE}^{\text{real}} \approx_\varepsilon \rho_{KE}^{\text{ideal}}$ where $\rho_{KE}^{\text{ideal}} = \frac{\mathbb{I}_K}{2^m} \otimes \rho_E$.

To achieve such a key distribution protocol, we will consider the following communication channels which Alice and Bob may have access to:

1. A classical channel: Alice and Bob can send classical bits in either direction over this channel. Eve has complete access to this channel. In particular, she can read all messages, modify them, and even impersonate Alice (or Bob).
2. A classical *authenticated* channel (CAC): A classical communication channel with one extra

guarantee: Alice and Bob know that the message originated unaltered from Alice or Bob respectively. This means that while the channel is not secret because Eve can still read all the messages that travel across, she cannot impersonate Alice or Bob or alter messages traveling over the channel.

3. A classical *secret* channel: A classical communication channel in which Eve cannot learn any information (see below!) about the messages traveling across. Yet, while she cannot hope to gain any information about the message, Eve could impersonate Alice or Bob.
4. A classical *secret and authenticated* channel: A classical communication channel combining both guarantees above.
5. A *quantum communication* channel: A channel where Alice may send quantum information (in particular, in the form of qubits) to Bob, where Eve has full access to all the quantum communication.

For simplicity we will start our discussion by assuming that Alice and Bob are already connected by a CAC - we will see later how such a CAC can be built. That is, we will for the moment only worry about establishing a key which is hidden from Eve!

5.2 Distributing keys given a special classical channel

As a warmup, let us consider how we can generate a key from a very special classical channel - which will include many of the essential ideas we will need in quantum key distribution. This special channel has the property that Eve cannot completely intercept all messages going across, but her ability to eavesdrop is somehow *guaranteed* to be limited. For the moment, let us take the special channel to have the following properties: whenever Alice sends a bit $b \in \{0,1\}$ across the special channel (SC) then

- Bob correctly receives b.
- Eve obtains the bit b correctly with probability q, otherwise with probability $1-q$ receives the bit $1-b$ (but she does not know whether the bit is correct or not).

In this special channel, we have modelled noise for Eve's infomation in an explicit manner, where each bit of b^E is obtained from b by flipping it with some probability $1-q$. Such a noise model is called the binary symmetric channel, which we denote as $BSC(q)$.

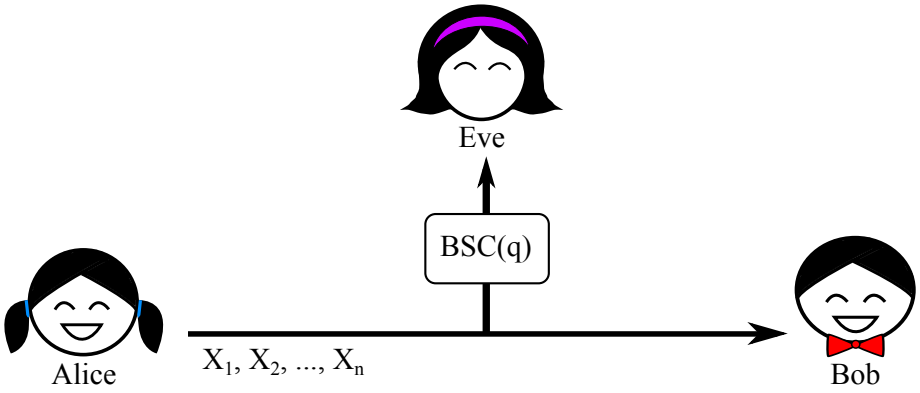

Figure 5.1: Distributing keys over a special classical channel.

Let us now consider the following simple protocol.

Protocol 1 — Key distribution using a special channel. Consider the following protocol:

1. Alice chooses a string $x = x_1, \ldots, x_n \in \{0,1\}^n$ uniformly at random, and sends each bit x_j

to Bob over the special channel.
2. Alice picks a random seed $r \in \{0,1\}^m$, uses a function $Ext : \{0,1\}^n \times \{0,1\}^m \to \{0,1\}^l$ and applies the extractor to X, computing $k = Ext(x,r)$.
3. Alice sends r to Bob over the CAC.
4. Bob computes $k = Ext(x,r)$.

Remember that to show that our protocol works we need to establish two things: first, we want that the protocol is ε-correct, that is, Alice and Bob output the same key (except for some small probability of failure). Second, we want to show that the protocol is ε-secure, for some parameter ε. To see that the protocol is correct, note that the special channel is such that Bob receives all bits correctly. That is, he obtains $x = x_1, \ldots, x_n$ without error. Since he also learns r, i.e., he knows which function $Ext(\cdot, r)$ to apply to x, he can compute $k = Ext(x,r)$ without error.

Why would this protocol be secure? Let us first note that Eve's probability of guessing each bit is precisely given by q. We thus have that

$$P_{\text{guess}}(X|E) = q^n \ , \tag{5.1}$$

or equivalently

$$H_{\min}(X|E) = -\log P_{\text{guess}}(X|E) = n(-\log q) \ . \tag{5.2}$$

If r is, for example, used to choose a function $Ext(\cdot, r)$ from a set of two-universal functions, then we are guaranteed that

$$D\left(\rho_{KRE}, \frac{\mathbb{I}}{2^\ell} \otimes \rho_{RE}\right) \leq \varepsilon \tag{5.3}$$

whenever $\ell \leq H_{\min}(X|E) - 2\log(1/\varepsilon) - 1$, as shown in [Ren05]. We have also seen this in the lecture notes last week. Therefore, whenever we generate a key that is at most $\ell \leq n(-\log q) - 2\log(1/\varepsilon) - 1$ bits long with this procedure, then we know that we are ε-secure. In cryptography, we typically fix ε and ℓ in advance, and then ask how large n has to be in order to achieve the desired key length ℓ with the guarantee ε. Note that Eq. (5.3) is to some extend remarkable: even if the eavesdropper *later* learns which function $Ext(\cdot, r)$ we applied, then nevertheless she cannot learn anything about the key! The fact that she learns r only later, however, is of crucial importance. If Eve would know r ahead of time, she could tailor her entire attack to the knowledge of r. In the protocol above this clearly wouldn't matter much at all, since we are given such a strong guarantee on what Eve can do in order to learn the bits, but in general we will have to be content with knowing only the min-entropy without such additional information.

We have seen now that whenever it is guaranteed that Eve cannot intercept all messages perfectly, i.e. there are some noise or losses for her while receiving information from Alice, then the min-entropy $\mathrm{H}_{\min}(X|E)$ will be high (since it is additive over n rounds), and therefore if Bob receives X perfectly, then Alice and Bob can always extract a key which is ε-secure against Eve.

■ **Example 5.2.1** Consider another special channel where Eve receives bits without noise, however Eve has limited memory and hence can only store a maximum number of S bits, that is, $|E| \leq 2^S$. If Alice sends a completely random, n-bit string X across the channel, then $\mathrm{H}_{\min}(X) = n$, and Eve's knowledge about X can be lower bounded by

$$\mathrm{H}_{\min}(X|E) \geq \mathrm{H}_{\min}(X) - \log|E| \geq n - S. \tag{5.4}$$

We thus see that whenever the length of X is greater than Eve's storage, i.e. $n > S$, Alice and Bob can use an extractor to extract a non-zero amount of key which will be secure against Eve. ■

5.3 Information reconciliation

So far, we have always assumed that there are no errors on the channel connecting Alice and Bob. Clearly, this is extremely unrealistic in any real implementation. If we follow the procedure in Protocol 1 when using this channel, the *correctness* of the protocol is affected: Alice and Bob now hold two different strings $X, \tilde{X}$ which are different from each other. Therefore, while performing privacy amplification, they will hash down two different strings and hence (very likely!) end up with two different keys $K_A \neq K_B$ that are useless for further communication.

How can we deal with this problem? The key idea is to perform error-correction. To this end, Alice and Bob need to perform one additional step before privacy amplification called *information reconciliation* in which they exchange error-correcting information to correct errors.

First of all, let's describe the communication scenario and introduce some convenient notation. Alice and Bob hold two strings that we denote X_A and X_B. X_B, the string of Bob, equals X_A plus a string of errors that we denote by S. Alice and Bob are connected by a CAC, and therefore an information reconciliation protocol consists simply in the exchange of messages over this channel in order for Bob to recover X_A. We denote by C the string consisting of all the messages exchanged over the classical channel. Finally, Bob, with the help of X_B and C will output $\hat{X}_A$, that is, an estimate of the string of Alice X_A.

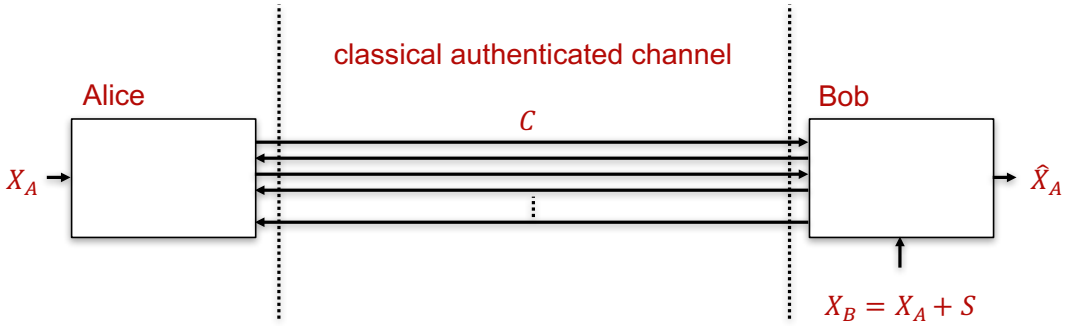

Figure 5.2: Scheme of a generic information reconciliation protocol.

Definition 5.3.1 — Information Reconciliation. Let X_A, X_B be distributed according to the joint probability distribution $P_{X_A X_B}$. An *information reconciliation* protocol for X_A, X_B is ε-correct and leaks $|C|$ bits if :

- $\text{Prob}(X_A \neq X_B) \leq \varepsilon$.
- The length of the messages exchanged on the public channel is $|C|$.

The goals of the information reconciliation step are two-fold. First and most obvious, Alice and Bob want to ensure that after reconciliation the strings X_A and $\hat{X}_A$ are ε-correct, that is, that the probability that they are different is at most ε. Second, note that all classical communication between Alice and Bob is public, which means that Eve can also gain information from the error correction information they send across! Again recalling the chain rule for the min-entropy, we have

$$H_{\min}(X|EC) \geq H_{\min}(X|E) - |C| \,, \tag{5.5}$$

where we used C to denote the error-correcting information sent across the classically authenticated channel. We thus have the min-entropy of Eve *with* the error-correction information C can shrink by at most the number of bits $|C|$ of error-correction information that Alice and Bob send.

Again, it is very easy to achieve any of both goals independently. Why is this? Imagine that a reconciliation protocol consists of Alice sending her whole string to Bob over the classical channel.

This is a great protocol if we only care about correctness, the strings will definitely be correct, but the leakage is maximal and after reconciliation we would not have any min-entropy left to do privacy amplification.

On the other hand, imagine a reconciliation protocol that consists in Alice and Bob doing nothing, then, for leakage purposes, the protocol is perfect, the leakage is zero, but the strings might not be correct for the desired ε.

We can classify information reconciliation protocols depending on their usage of the classically authenticated channel. The most general protocol might consist in the exchange of messages in both directions, that is from Alice to Bob and from Bob to Alice. We call such a protocol a two-way or an interactive protocol. However, much more simpler, and in many circumstances, it is already sufficient that the whole reconciliation consists of a single message from Alice to Bob, that is, the communication happens one-way. We refer to such protocols as one-way reconciliation protocols, where Alice encodes her string X_A into a single (significantly shorter) message that we denote by C_A and sending it through the classical channel. Then Bob uses C_A to eliminate errors from X_B, effectively recovering X_A.

5.4 Syndrome coding

In the following we will explore one concrete one-way reconciliation scheme. The scheme that we will describe is based on linear codes. Let us review the definition and main elements of linear codes:

Definition 5.4.1 — Linear code. Let $\mathbb{F}_q$ be the finite field of size q, a (n,k) q-ary linear code C is a linear subspace of $\mathbb{F}_q^n$ of dimension k.

The individual elements (which are q-ary strings of length n) contained in the subspace defining a linear code are called *codewords*, and a (n,k) q-ary code has q^k different codewords. We will only be concerned with binary codes, so in the following we let $q = 2$.

Since a code is just a subspace, if we want to construct an n length binary code, any procedure that characterizes a subspace of $\mathbb{F}_2^n$ will allow us to construct a code. One convenient characterization is by using a $m \times n$-dimensional *parity check matrix* H. The code is defined as the set of vectors v such that $H \cdot v^T = 0$. The dimension of the code induced by H is $k = n - \text{rank}(H)$.

The map $s_H : \mathbb{F}_2^n \to \mathbb{F}_2^m$ given by $v \mapsto H \cdot v^T$ is called the *syndrome*. It turns out that the syndrome is very useful beyond inducing the code. In particular, an example of a reconciliation procedure is when Alice and Bob agree on a particular parity matrix H. Let's consider this in more detail. First, let us discuss the encoding step performed by Alice. The reconciliation scheme is based on linear codes, given a parity check matrix H and Alice's vector X_A, the encoding of X_A is its syndrome. That is, the message that Alice sends to Bob for information reconciliation is the syndrome of X_A.

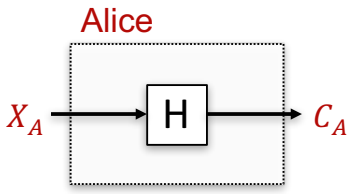

Figure 5.3: The encoder in syndrome coding based reconciliation.

■ **Example 5.4.1** Let $v = (001)$ and $H = \begin{pmatrix} 110 \\ 011 \end{pmatrix}$. Then $s_H(v) = H \cdot v^T = (01)^T$. What this means,

is that if v were the string of Alice, the message that Alice would send to Bob for reconciliation would be $s_H(v)$ which as we calculated is: $(01)^T$. ∎

Let us now move to Bob, who has a decoder, which is essentially a procedure that helps him to estimate the error string S (where recall $X_B = X_A + S$), so that he may recover X_A. The decoder is a little bit more complicated than the encoder, since it takes both C_A and X_B as inputs. The first thing that happens in the decoder is that it computes the syndrome of X_B that we call C_B. Then, Bob computes $C_S = C_B + C_A$, where recall that C_A is the syndrome of X_A. We call this resulting string C_S because it is indeed the syndrome of the error string, i.e. $C_S = s_H(S)$. Then, C_S is sent into a module that estimates the error string S and outputs the estimate that we call $\hat{S}$. Finally $\hat{S}$ is added to X_B and this will be the decoded string that Bob will receive.

Exercise 5.4.1 Show that C_S is indeed the syndrome of the error string S, or in other words, that $s_H(S) = s_H(X_A) + s_H(X_B)$. ∎

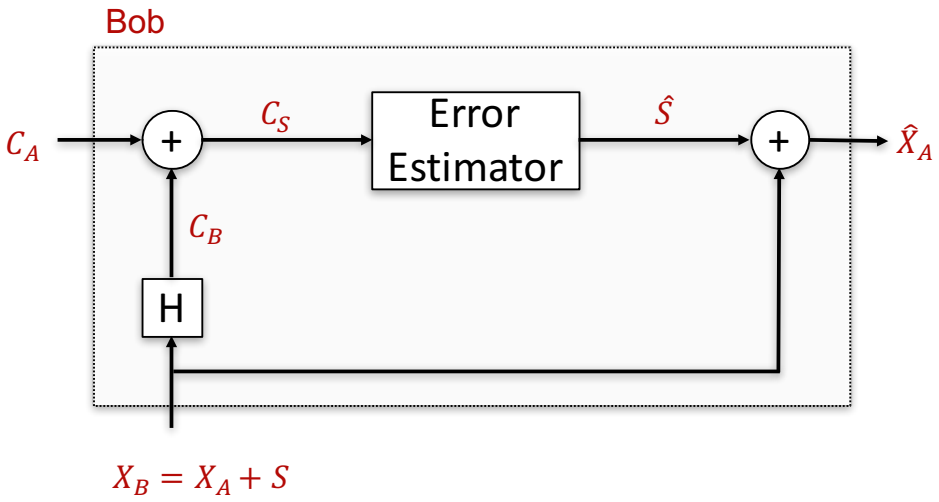

Figure 5.4: The decoder in syndrome coding based reconciliation.

Exercise 5.4.2 Show that if the error estimate is correct $\hat{X}_A$ will equal X_A. ∎

A simple but relevant property of the scheme is that as soon as the length of the string that Alice sends Bob is smaller than the length of the string of Alice not all errors can be corrected. In order to see this recall that $\hat{X}_A$ is $X_B + \hat{S}$, but $\hat{S}$ is a function of C_S, the syndrome of S. Hence, even if the estimator function outputs a different value for each syndrome we have 2^m different outputs while there are 2^n different error strings, in other words, unless m equals n, it is not possible to correct all errors. However, if different errors occur with different probabilities we might be satisfied if we correct the most likely errors.

∎ **Example 5.4.2** Let us go back to Example 5.4.1. Let us now describe the decoder, for this we need to make explicit the estimation function. There are four different syndromes. We will assign as error estimate for each syndrome the following strings:

Syndrome	Error Estimate
00	000
01	001
10	100
11	010

Can you guess why we chose this particular map? The idea is that if there is zero or one errors, the estimator will output the correct error estimate. However, this also means that any other error string will be wrongly estimated. ∎

5.5 Limits of reconciliation

We have seen a concrete scheme for reconciliation. What are the fundamental limits of reconciliation? In order to answer this question we need some structure. Let us assume that the strings of Alice and Bob, that we denote for precision X_A^n and X_B^n, are of length n, where each of the symbols is drawn independently from the same joint distribution $P_{X_A X_B}$ (where X_A, X_B are binary random variables). Then, any information reconciliation protocol that leaks $|C|$ bits satisfies:

$$|C| \geq n \cdot H(X_A|X_B) \tag{5.6}$$

Moreover, the inequality can be achieved when $n \to \infty$ [SW73].

In a realistic scenario n is finite and the information reconciliation protocol needs to be computationally efficient. Instead of dealing with the implementation details of an information reconciliation protocol, it is sometimes convenient to approximate the leakage value of a realistic protocol by $\xi \cdot nH(X_A|X_B)$, where $\xi > 1$ is the reconciliation efficiency. The constant ξ is often chosen $\xi \approx 1.2$. However, this approximation should be used with care since ξ will be a function of the length, the noise model and the correctness considered [Tom+14].

The errors between Alice's and Bob's string can also generally be modelled by a BSC. That is, we can see their strings as the input and output of a BSC: whenever Alice inputs a bit x, Bob receives x with probability p, but with probability $1-p$ the bit is flipped to $x+1$[1]. In the case of a BSC, Eq. (5.6) simplifies to $|C| \geq nh(p)$ where $h(p) = -p\log(p) - (1-p)\log(1-p)$ is the binary entropy function.

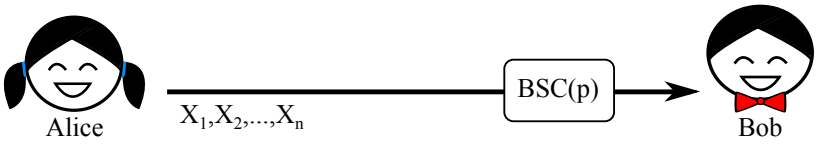

Figure 5.5: The errors between Alice's and Bob's string are generally modelled by a BSC.

5.6 Further reading

So let us conclude this module on reconciliation with some considerations.

The example that we described in Examples 5.4.1 and 5.4.2, works but it does not scale up, if we try to extend the same ideas to a parity check matrix of m by n, we will need a table with 2^m entries to decide the output of the estimator module. However, the method of reconciliation based

[1] In the context of quantum key distribution (QKD), this error probability $1-p$ is also often called the quantum bit-error rate (QBER).

on syndrome can be adapted to use any family of linear codes. Previous work has obtained leakage close to the theoretical optimal using LDPC codes, polar codes, turbo codes, etc.

Previous to this idea of using linear error correcting codes and one-way reconciliation, there was some ad-hoc two-way protocol, proposed specially for information reconciliation. Its name is Cascade [BS93], it has a reasonably simple description and it is not very difficult to implement. However, the original version was suboptimal from the point of view of leakage. Fortunately, there are some nice modifications that make it competitive with the error correcting codes based reconciliation.

Acknowledgments

 This chapter of the lecture notes was written by David Elkouss, Nelly Ng, Thomas Vidick and Stephanie Wehner. We thank Kenneth Goodenough, Jonas Helsen, Jérémy Ribeiro, and Jalex Stark for proofreading.

Bibliography

[BS93] Gilles Brassard and Louis Salvail. “Secret-key reconciliation by public discussion”. In: *Workshop on the Theory and Application of of Cryptographic Techniques*. Springer. 1993, pages 410–423 (cited on page 10).

[Ren05] R. Renner. “Security of Quantum Key Distribution”. PhD thesis. ETH Zurich, 2005 (cited on page 5).

[SW73] David Slepian and Jack Wolf. “Noiseless coding of correlated information sources”. In: *IEEE Transactions on information Theory* 19.4 (1973), pages 471–480 (cited on page 9).

[Tom+14] Marco Tomamichel et al. “Fundamental finite key limits for information reconciliation in quantum key distribution”. In: *2014 IEEE International Symposium on Information Theory*. IEEE. 2014, pages 1469–1473 (cited on page 9).

Lecture Notes

Quantum Cryptography Week 6:

Quantum key distribution protocols

Contents

Last week we saw the definition of a correct and secure key distribution protocol. A *quantum key distribution* (QKD) protocol allows Alice and Bob to harness the advantages of quantum information processing to generate a shared secret key. Wiesner already suggested a quantum key distribution protocol in the 70s [Wie83]. The most well known, and indeed the oldest QKD protocol with a name is called BB84, after the inventors Bennett and Brassard [BB84].

This week we focus on the BB84 protocol. It turns out that for this protocol it is enough for us to prepare and measure single qubit quantum states. Let us thus imagine that Alice and Bob are connected by a quantum channel: unlike quantum computing, quantum cryptography is feasible *today*, at least over short distances, and it is reasonable to assume that Alice and Bob can transmit qubits over an optical fiber.

6.1 BB84 Quantum key distribution

In last week's lectures we discussed a special classical channel, where Eve is guaranteed to have some amount of noise in her attempts at intercepting bits transmitted by Alice to Bob. In such a classical protocol, if one takes away the guarantee about Eve but instead allow her to arbitrarily intercept messages on the special channel, then it is clear that there is no more security: Eve can learn all the bits of the string x. When considering a quantum channel, the classical protocol would amount to sending a string x encoded in a single fixed basis. For example, we might send x in the standard basis as $|x\rangle\langle x| = |x_1\rangle\langle x_1| \otimes \ldots \otimes |x_n\rangle\langle x_n|$. Eve, knowing the basis, can measure the transmitted quantum state to recover $x_1 \cdots x_n$ without error, copying each bit without causing any disturbance to the state. Of course, we might also encode x in a different basis, for example the Hadamard basis, as $H^{\otimes n}|x\rangle\langle x|H^{\otimes n}$. Yet, the fact remains, if Eve knows the basis, then she can copy the bits without being detected!

Exercise 6.1.1 Consider a bit b encoded in the Hadamard basis $H|b\rangle\langle b|H$. Give a measurement that recovers b (knowing it was encoded in the Hadamard basis!). Compute the post-measurement states for each possible outcome. What do you conclude? ∎

However, recall that by the no-cloning theorem presented in the Week 0 lecture notes, it is impossible to copy arbitrary qubits, i.e., qubits that could live anywhere on the Bloch sphere. This is precisely the case when Eve does not know the encoding in advance. We are thus motivated to let Alice not just choose bits x_j at random, but for each bit she will also randomly choose a basis θ_j. This gives rise to the BB84 encoding:

Definition 6.1.1 — BB84 states/encoding. The BB84 states are $\{|0\rangle, |1\rangle, |+\rangle, |-\rangle\}$. This set of states corresponds to encoding a classical bit $x_j \in \{0,1\}$ in a randomly chosen basis $\theta_j \in \{0,1\}$ where $\theta_j = 0$ labels the standard basis, and $\theta_j = 1$ the Hadamard basis.

Note that the standard basis and the Hadamard basis are the eigenbases of the Pauli-Z and Pauli-X matrices respectively. Another similar set of states is known as the six-state encoding, which consists the eigenbases of the Pauli matrices X, Y and Z.

Definition 6.1.2 — Six-state encoding. The six-states are $\{|0\rangle, |1\rangle, |+\rangle, |-\rangle, |+y\rangle, |-y\rangle\}$, where

$$|\pm y\rangle = \frac{1}{\sqrt{2}}(|0\rangle \pm i|1\rangle). \tag{6.1}$$

This set of states corresponds to encoding a classical bit $x_j \in \{0,1\}$ in a randomly chosen basis $\theta_j \in \{0,1,2\}$, where $\theta_j = 0$ labels the standard basis, $\theta_j = 1$ the Hadamard basis, and $\theta_j = 2$ represents the eigenbasis of the Pauli Y matrix.

Both the BB84 basis and six-state basis are used frequently in quantum cryptographic protocols. This week, we will consider how BB84 states are used in a quantum key distribution protocol, focusing first on the case of noiseless transmission, i.e. the quantum channel is the identity channel, it transmits the quantum state without any errors. We first assume that Alice and Bob are connected via an classical authenticated channel (CAC), which they will use during the protocol. Later in the notes we will investigate how Alice and Bob could construct such a channel.

The BB84 protocol can be described as follows:

Protocol 1 — BB84 QKD (no noise). Outputs $k \in \{0,1\}^\ell$ to both Alice and Bob. Alice and Bob execute the following:

1. For a small real-valued parameter $\eta \ll 1$, and a large integer $n \gg 1$, Alice chooses a string $x = x_1, \ldots, x_N \in \{0,1\}^N$ uniformly at random where $N = (4+\eta)n$. She also chooses a basis string $\theta = \theta_1, \ldots, \theta_N$ uniformly at random. She sends to Bob each bit x_j by encoding it in a quantum state according to the basis θ_j: $H^{\theta_j}|x_j\rangle$.
2. Bob chooses a basis string $\tilde{\theta} = \tilde{\theta}_1, \ldots, \tilde{\theta}_N$ uniformly at random. He measures qubit j in the basis $\tilde{\theta}_j$ to obtain outcome $\tilde{x}_j$. This gives him a string $\tilde{x} = \tilde{x}_1, \ldots, \tilde{x}_N$.
3. Bob tells Alice over the CAC that he has received and measured all the qubits.
4. Alice and Bob tell each other over the CAC their basis strings θ and $\tilde{\theta}$ respectively.
5. Alice and Bob discard all rounds j in which they didn't measure in the same basis. Let $S = \{j | \theta_j = \tilde{\theta}_j\}$ denote the indices of the rounds in which they measured in the same basis. Since Alice and Bob chose $\theta, \tilde{\theta}$ at random, for large values of n, they throw away roughly $N/2 \approx 2n$ bits.
6. Alice picks a random subset[a] $T \subseteq S$ for testing and tells Bob T over the CAC. That is, Alice and Bob test roughly $|T| \approx N/4 \approx n$ bits.
7. Alice and Bob announce x_T and $\tilde{x}_T$ to each other over the CAC, where we denote by x_T the substring of x corresponding to the indices in the test set T. They compute the error rate $\delta = W/|T|$, where $W = |\{j \in T \mid x_j \neq \tilde{x}_j\}|$ is the number of errors when Alice and Bob did measure in the same basis.
8. If the error rate $\delta \neq 0$, then Alice and Bob abort the protocol. Otherwise, they proceed to denote $x_{\text{remain}} = x_{S \setminus T}$ and $\tilde{x}_{\text{remain}} = \tilde{x}_{S \setminus T}$ as the remaining bits, i.e., the bits where Alice and Bob measured in the same basis, but which they did not use for testing. The length of x_{remain} and $\tilde{x}_{\text{remain}}$ is approximately n bits.
9. Alice and Bob perform privacy amplification: Alice picks a random r, and computes $k = Ext(x_{\text{remain}}, r)$. She sends r to Bob, who computes $k = Ext(\tilde{x}_{\text{remain}}, r)$.

[a] A random subset T of S is where each element in S is included in T with probability $1/2$. By this definition, if $|S|$ is large, then $|T| \approx |S|/2$.

Note that even though steps 1 and 2 seem to take place one after the other, and you may be tempted to think that Alice and Bob require quantum storage, this is not necessarily the case. Alice can prepare the qubits one-by-one, and Bob can also measure them one-by-one. This is very appealing, since Alice and Bob only need very simple quantum devices - preparing and measuring single qubits is already enough!

Let us first investigate why the protocol is correct. If there are no errors in transmission, then whenever Bob measures in the same basis as the one chosen by Alice ($\theta_j = \tilde{\theta}_j$), then he learns the bit perfectly ($x_j = \tilde{x}_j$). If there is no eavesdropper, they will pass the test. Since $x_{\text{remain}} = \tilde{x}_{\text{remain}}$ and Bob knows r they produce the same output k as before.

But why should this protocol be secure? The intuition is that whenever Eve tries to intercept, and gain some information from the transmitted qubits, she will invariably disturb the quantum states — and Alice and Bob can detect such disturbance. It can be proven that

$$H_{\min}(X_{\text{remain}}|E) \gtrsim n\,[1 - h(\delta)]\ , \tag{6.2}$$

where $h(\delta) = -\delta \log \delta - (1-\delta)\log(1-\delta)$ is the binary entropy function, and recall from Protocol 1 that δ is the error rate between Alice and Bob's sample. We note that analyzing the sampling procedure, i.e., which qubits to test, for small values of N is an intricate problem which requires great care, as analyzed in [Pfi+16]. Here we will not dive into this but instead consider only the limit of large N.

In the case where errors in transmission occur, the error rate is always $\delta \neq 0$. This affects the correctness of Protocol 1, since Alice and Bob will get $x_{\text{remain}}, \tilde{x}_{\text{remain}}$ respectively, where $x_{\text{remain}} \neq \tilde{x}_{\text{remain}}$. In fact, when $x_{\text{remain}} \neq \tilde{x}_{\text{remain}}$, by property of the extractor Ext, with probability almost equal to 1, $Ext(x_{\text{remain}}, r) \neq Ext(\tilde{x}_{\text{remain}}, r)$. This means that almost for certain, Alice and Bob will end up with different keys!

To overcome this problem, Alice and Bob will have to perform an additional step of *information reconciliation* in the protocol. Thus instead of Protocol 1, they execute the following protocol:

Protocol 2 — BB84 QKD (with noise). Outputs $k \in \{0,1\}^\ell$ to both Alice and Bob.
Alice and Bob execute the following:

1-7. Same as Protocol 1.

8. If the error rate is larger than a certain threshold $\delta > \delta_t$, Alice and Bob abort the protocol. Otherwise, they proceed to denote $x_{\text{remain}} = x_{S\setminus T}$ and $\tilde{x}_{\text{remain}} = \tilde{x}_{S\setminus T}$ as the remaining bits, i.e., the bits where Alice and Bob measured in the same basis, but which they did not use for testing.
9. Alice and Bob perform information reconciliation: Alice sends some error correcting information C across the classical authenticated channel to Bob, and Bob corrects the errors in his string $\tilde{x}_{\text{remain}}$, so that he can obtain x_{remain} from the process as well.
10. Alice and Bob perform privacy amplification: Alice picks a random r, and computes $k = Ext(x_{\text{remain}}, r)$. She sends r to Bob, who computes $k = Ext(\tilde{x}_{\text{remain}}, r)$.

In Protocol 2, Alice and Bob allow for errors under the assumption that all the errors can be caused by a malicious Eve. They bound the amount of min-entropy Eve has about X_{remain} by (i) first invoking Eq. (6.2), and (ii) taking into account the amount of error correction information C sent across the channel from Alice to Bob. The size of k (given by the number of bits l) then depends on both δ and $|C|$.

Later on we will use the guessing game from last week in order to prove that the BB84 protocol presented aboveis secure for certain positive values of l.

6.2 Security of BB'84

To prove security of the BB'84 protocol we make two small modifications to the protocol. Although it will at first appear like these modifications give more power to the eavesdropper, they will facilitate the analysis.

6.2.1 A purified protocol

The first modification is rather benign. Consider the following two experiments. In the first experiment, Alice chooses $x, \theta \in \{0,1\}$ uniformly at random and returns $|x\rangle_\theta = H^\theta |x\rangle$, an encoding of the bit x in the basis specified by θ (the standard basis if $\theta = 0$ and the Hadamard basis if $\theta = 1$). In the second experiment, Alice first prepares an EPR pair $|\phi^+\rangle = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$. She then chooses a $\theta \in \{0,1\}$ uniformly at random and measures the first qubit in the basis $\{|0\rangle_\theta, |1\rangle_\theta\}$, obtaining outcome $x \in \{0,1\}$. She returns the second qubit.

We claim that the two experiment are absolutely equivalent. There are two things to verify. First, while in the first experiment Alice makes a choice of x uniformly at random, in the second experiment x is determined as the outcome of a measurement on the EPR pair. But we know that,

since the reduced density matrix of the EPR pair on the first qubit is the totally mixed state, any basis measurement on that qubit will return each of the two possible outcomes with probability $1/2$. So the distribution of x is identical in the two experiments.

Second, we should check that when Alice obtains outcome x by measuring the first qubit of the EPR pair in the basis θ, the qubit she returns, i.e. the second qubit of the EPR pair, is indeed projected onto the state $|x\rangle_\theta$. Again this is a property of the EPR state that is valid for any choice of basis measurement on the first qubit, so we are good — the two experiments are indeed equivalent.

Let us then consider an equivalent formulation of the BB'84 protocol in which, instead of directly preparing BB'84 states, Alice first prepares EPR pairs, keeps the first qubit of each pair to herself, and sends the second qubit to Bob. At a later stage she measures her qubit in a basis $\theta_j \in \{0,1\}$ chosen uniformly at random, and records the outcome x_j.

Thanks to the observation we made above this new formulation of the protocol is completely equivalent to the standard one. Even though it may look more complicated, the essential advantage of the new formulation is that it allows us to delay the moment in the protocol at which Alice needs to make her choice of basis. Although the difference is only conceptual, we can think of this delay as giving less power to Eve: we will now be able to easily argue that certain actions of the eavesdropper, taken early on in the protocol, could not have depended on Alice's basis choice, since the choice has not yet have been made at the time.

Here is the modified protocol in detail. For simplicity we again consider the case where there is no noise. It is called the "purified" BB'84:

Protocol 3 — Purified BB'84 (no noise).. Outputs $k \in \{0,1\}^\ell$ to both Alice and Bob.

1. For a small real-valued parameter $\eta \ll 1$, and a large integer $n \gg 1$, let $N = (4+\eta)n$. Alice prepares N EPR pairs $|\phi^+\rangle_{AB}$, and sends the second qubit of each pair to Bob. She chooses a uniformly random basis string $\theta = (\theta_1, \ldots, \theta_N) \in \{0,1\}^N$ and measures each of her qubits in the bases θ to obtain a string $x = x_1, \ldots, x_N$.
2. Bob chooses a uniformly random basis string $\tilde{\theta} = (\tilde{\theta}_1, \ldots, \tilde{\theta}_N) \in \{0,1\}^N$. He measures the j-th qubit he received from Alice in the basis $\tilde{\theta}_j$ to obtain outcome $\tilde{x}_j$.
3. Bob tells Alice over the CAC that he received and measured all the qubits.
4. Alice and Bob exchange their basis strings θ and $\tilde{\theta}$ over the CAC.
5. Alice and Bob throw away the data from all rounds $j \in \{1, \ldots, N\}$ in which they didn't measure in the same basis. Let $S = \{j | \theta_j = \tilde{\theta}_j\}$ denote the indices in which they measured in the same basis.
6. Alice picks a random subset $T \subseteq S$ of size $T \approx |S|/2$ for testing and tells Bob T over the CAC.
7. Alice and Bob announce x_T and $\tilde{x}_T$ to each other over the CAC. They compute the error rate $\delta = W/|T|$, where $W = |\{j \in T \mid x_j \neq \tilde{x}_j\}|$ is the number of disagreements found in T. If δ is too large, they abort the protocol.
8. Let $R = S \backslash T$. Alice and Bob perform information reconciliation and privacy amplification on x_R.

The idea of considering a purified variant of the BB'84 protocol can be traced back to a different proposal for quantum key distribution put forward by Ekert in 1991 [Eke91]. Ekert's main insight was that if Alice and Bob were able to test for the presence of entanglement between their qubits, then (intuitively) by the monogamy of entanglement they would be able to certify that their systems are uncorrelated with Eve's. We will explore Ekert's protocol (and prove the intuition correct!) next week when we analyze quantum key distribution in the so-called device-independent setting.

Even though the purified protocol requires Alice to prepare EPR pairs, the formulation will only be used for the purposes of analysis. As we already discussed, from the point of view of

any eavesdropper which protocol Alice and Bob actually implement makes no difference at all, so it is perfectly fine to prove security of the purified protocol but use the original BB'84 protocol in practice. This is convenient because it is much easier to prepare single-qubit BB'84 states than to distribute EPR pairs across long distances.

6.2.2 More power to the eavesdropper

The second modification we make to the BB'84 is less benign, and will appear to give much more power to the eavesdropper. But once again it will be convenient for the analysis. Moreover, if we can prove security against stronger eavesdroppers without too much extra effort, why not do it?

The motivation for this second modification is that it is very hard to model the kinds of attacks Eve might apply to the quantum communication channel between Alice and Bob. For example, she might partially entangle herself with the qubits sent by Alice, creating a joint state ρ_{ABE} on which we have little control.

Exercise 6.2.1 Consider the case of a single EPR pair ($n = 1$), and suppose that Eve applies a CNOT on her qubit $|0\rangle_E$, controlled on the qubit B that Alice sends to Bob (Eve then forwards the qubit over to Bob). Compute the resulting joint state ρ_{ABE}. Compute the probability that Alice and Bob choose the same basis $\theta = \tilde{\theta}$ and obtain $x = \tilde{x}$. Is this a good attack? ∎

Because it is hard to model general intercepting attacks of the form described in the exercise, we will modify the protocol by allowing Eve to prepare an arbitrary pure state ρ_{ABE}, where the A and B systems are each made of N qubits, then give A to Alice, B to Bob, and keep E to herself. Alice and Bob will each measure their respective qubits using random choices of bases as in the protocol, and proceed from there on. By giving more power to Eve (she prepares the states, instead of Alice) we're preventing ourselves from thinking too hard about having a model for the attacks: in the new setup, Eve can prepare any state she likes!

This may sound crazy: if we let the eavesdropper prepare any state, then why doesn't she choose, say, $\rho_{ABE} = |000\rangle_{ABE}^{\otimes N}$? Observe that such a state would pass the "matching outputs" test when $\theta_j = \tilde{\theta}_j = 0$ (standard basis), but it would completely fail whenever $\theta_j = \tilde{\theta}_j = 1$ (Hadamard basis). So even though we're claiming Eve could prepare any state she likes, not all states will be accepted by Alice and Bob in the "matching outputs" test they perform in Step 7. How powerful is this test? Can it be used to certify that the state handed over by Eve indeed has the correct form, of being (close to) a tensor product of N EPR pairs? This may sound surprising, as the test only involves local measurements: can local measurements really detect entanglement? The answer is yes. Let's see how it works.

6.2.3 Locally implementing an entangled measurement

Suppose we modified the purified BB'84 protocol by adding an initial step as follows:

0. *Upon receiving their N respective qubits from Eve, Alice and Bob jointly measure each pair of qubits using the two-outcome POVM* $\{|\phi^+\rangle\langle\phi^+|_{AB}, \mathbb{I}_{AB} - |\phi^+\rangle\langle\phi^+|_{AB}\}$*, where* $|\phi^+\rangle_{AB}$ *denotes the EPR pair on Alice Bob's joint system. If the number of pairs of qubits that were not found to equal* $|\phi^+\rangle_{AB}$ *is larger than* δn *they abort Protocol 3. Otherwise, they proceed as usual.*

With this modification the protocol is clearly secure, and the tests performed in step 7 have become superfluous. Indeed, after the completion of step 0, Alice and Bob already have the guarantee that at least $(1-\delta)n$ of their shared pairs of qubits are perfect EPR pairs (since they are projected in the post-measurement state $|\phi^+\rangle$). In particular, any bit of the raw key obtained from measurements on these states is perfectly uniform and uncorrelated with Eve (remember that correlations generated from pure states are always perfectly monogamous).

The problem with step 0 is that it requires Alice and Bob to perform a joint entangled measurement, which they cannot implement locally. Or can they?

Exercise 6.2.2 Suppose given a tripartite state ρ_{ABE}, where A and B are each systems of a single qubit. Show that the probability that a measurement of systems A and B in the standard basis results in matching outcomes is exactly $\mathrm{Tr}(\Pi_1 \rho_{AB})$, where

$$\Pi_1 = |\phi^+\rangle\langle\phi^+| + |\Psi_{01}\rangle\langle\Psi_{01}|, \qquad \text{and} \qquad |\Psi_{01}\rangle = \frac{1}{\sqrt{2}}|00\rangle - \frac{1}{\sqrt{2}}|11\rangle. \tag{6.3}$$

Similarly, show that if the measurement is performed in the Hadamard basis then the probability of obtaining matching outcomes is $\mathrm{Tr}(\Pi_2 \rho_{AB})$, with

$$\Pi_2 = |\phi^+\rangle\langle\phi^+| + |\Psi_{10}\rangle\langle\Psi_{10}|, \qquad \text{and} \qquad |\Psi_{10}\rangle = \frac{1}{\sqrt{2}}|01\rangle + \frac{1}{\sqrt{2}}|10\rangle. \tag{6.4}$$

■

The exercise suggests that the "matching outcomes" test that Alice and Bob implement in step 7 of the original Protocol 3 is essentially equivalent to its replacement step 0 introduced above. Therefore, the security of Protocol 3 with step 0 implemented should directly imply the security of the protocol without step 0, but with step 7 instead.

To express the relation between the two steps, consider the reduced density matrix $\rho_{A_jB_j}$ of the state prepared by Eve on any two qubits for Alice and Bob, and let p_j be the probability of succeeding in the "matching outcomes" test, averaged over the choice of a uniformly random (but identical for both A_j and B_j) basis in which to perform the test. Then by expressing $\rho_{A_jB_j}$ in the Bell basis as

$$\rho_{A_jB_j} = q_{00}|\phi^+\rangle\langle\phi^+| + q_{01}|\Psi_{01}\rangle\langle\Psi_{01}| + q_{10}|\Psi_{10}\rangle\langle\Psi_{10}| + q_{11}|\Psi_{11}\rangle\langle\Psi_{11}|, \tag{6.5}$$

using the expressions for Π_1 and Π_2 obtained in Exercise 6.2.2 you can check that the condition

$$q_{00} = \langle\Psi_{00}|\rho_{A_jB_j}|\Psi_{00}\rangle \geq 2p_j - 1 \tag{6.6}$$

is satisfied. In particular, if the probability of success in the test is close to 1, say $p_j = 1 - \delta$ for some small δ, then the overlap q_{00} is correspondingly large, at least $1 - 2\delta$.

If the test performed in step 7 of Protocol 3 was really equivalent to our hypothetical step 0 projecting all pairs of qubits on EPR pairs, then we would be done with our proof of security. However, although the intuition is valid the argument is not quite complete: the two tests are not *exactly* equivalent, and making the argument precise is going to require more work.

A first distinction is that, in Protocol 3, step 7 is performed on the rounds T selected for testing, whereas it is the rounds in $R = S\backslash T$ that are used for the raw key (the outputs used for the raw key are never tested for equality, as this would leak them to Eve!). Another difficulty is that the results of the tests performed in different rounds are not independent from each other: although Alice and Bob make independent measurements, the state ρ_{ABE} prepared by Eve does not necessarily have a tensor product form.

The second distinction raises a thorny difficulty, to which we'll return in more detail next week. For now, let's concentrate on the first objection: how do we infer conditions on the qubits in rounds $j \in S\backslash T$ from results of tests performed on the qubits in rounds $j \in T$?

6.2.4 A concentration inequality

Let us summarize the situation. Suppose for simplicity that the number $|S|$ of rounds in which Alice and Bob make the same basis choice is exactly $|S| = 2n$, and that T has size $|T| = |S|/2 = n$. For

each $j \in S$, introduce an indicator random variable $Z_j \in \{0,1\}$ such that $Z_j = 1$ indicates failure in the matching outcomes test: $Z_j = 0$ if and only if $x_j = \tilde{x}_j$. With this notation the condition verified by Alice and Bob at step 7 of Protocol 3 can be written as $\sum_{j\in T} Z_j \leq \delta|T|$. In order to analyze security of their key, however, they would like to bound $\sum_{j\in S\setminus T} Z_j$. How can we do this?

The key idea is to use the fact that T is chosen as a random subset. Intuitively the average number of failures in T should be about the same as the average in the whole of S: indeed, which rounds are included in T or not is chosen at random by Alice, independently from whether the outcomes in those rounds happened to match or not.

The main tool required to make this intuition precise is called a concentration bound. There are many such bounds. The most widely used are usually referred to as the "Chernoff bound" or "Hoeffding's inequality", which is a generalized version of the Chernoff bound. If you have never heard of them, go look them up! The following is a variant of the Chernoff bound that turns out to be perfectly tuned for our scenario:

Theorem 6.2.1 — Lemma 7 in (TL15). Let $m = n + k$ and consider binary random variables $X_1, \ldots, X_m$. (The X_i may be arbitrarily correlated.) Let T be a uniformly random subset of $\{1, \ldots, m\}$ of size k. Then for any $\delta, \nu > 0$,

$$\Pr\left(\sum_{j\in T} X_j \leq \delta k \ \wedge \sum_{j\in\{1,\ldots,m\}\setminus T} X_j \geq (\delta+\nu)n \right) \leq e^{-2\nu^2 \frac{nk^2}{(n+k)(k+1)}}. \tag{6.7}$$

To see what the theorem says in our setting, set $m = |S| = 2n$ and $k = n$. Let's also choose $\nu = \delta$ for convenience. Plugging in these parameters we get the bound

$$\Pr\left(\sum_{j\in T} Z_j \leq \delta n \ \wedge \sum_{j\in S\setminus T} Z_j \geq 2\delta n \right) \leq e^{-\delta^2 n}, \tag{6.8}$$

which is valid for any choice of $\delta > 0$. This bound implies that the probability that the test performed in step 7 passes, but the outcomes obtained in the non-tested rounds $R = S\setminus T$ do not match in a fraction larger than 2δ of these rounds, is tiny — exponentially small in n! Writing ABORT to denote the event that Alice and Bob abort in Step 7 of Protocol 3, we can use Bayes' rule to rewrite the bound above as

$$\Pr\left(\sum_{j\in S\setminus T} Z_j \geq 2\delta n \,\middle|\, \neg\text{ABORT} \right) \leq \frac{e^{-\delta^2 n}}{\Pr(\neg\text{ABORT})}. \tag{6.9}$$

Writing the bound in this way points to an important subtlety in how the security of quantum key distribution is defined. As you can see, the bound is only good if $\Pr(\neg\text{ABORT})$ is not too small; if this probability was extremely tiny, then the right-hand side of Eq. (6.9) would suffer a corresponding blow-up. The probability that the protocol does not abort is not something that we can control or test, and it is natural that this probability has to be taken into account when defining security: we should always allow the protocol to have a very small probability of not aborting, in which case no claim can be made on the security. We will see a precise definition for the security of quantum key distribution next week.

Unfortunately we are still not done, due to the issue of dependencies between different tests, which may arise due to the eavesdropper preparing a state that is not in tensor product form.

If the state ρ_{ABE} *is* a tensor product across different rounds, i.e. it is of the form $\rho_{ABE} = \otimes_{j=1}^{n} \rho_{A_jB_jE_j}$ then we can complete the proof. Using that the choice of basis $\theta_j, \tilde{\theta}_j$ for $j \in R$ is uniformly random, we can conclude from the bound in Eq. (6.9) that a large fraction of $j \in R$ are such that the state $\rho_{A_jB_j}$ *would* pass the matching outcomes test, in *both* bases, with high probability

(this is because, if it were not the case, there would be a sufficiently high chance that we make a choice of basis with respect to which the state fails the test, leading to a contradiction with Eq. (6.9)). From the analysis in Section 6.2.3 we can deduce that $\rho_{A_jB_j}$ has a correspondingly large overlap with an EPR pair, and thus that the outcomes obtained by Alice and Bob when measuring in the same basis are highly correlated with one another, but (due to monogamy) have very weak correlation with Eve's system. Working out the parameters will give us a bound on the min-entropy of X_j in each round $j \in R$, which can be added up over all rounds by using the independence of different rounds.

If the state ρ_{ABE} is not a tensor product, unfortunately the analysis becomes more difficult; for instance the min-entropy does not add up easily across rounds. We will give a detailed analysis under the independence assumption next week, and we will also discuss the non-independent case in greater detail.

6.3 Authentication

Let us return to an important assumption that is always made when considering the BB'84 protocol: that the communication channel between Alice and Bob, that we've been calling the "CAC", is what its name implies — an authenticated channel. This assumption is used to guarantee that, although the eavesdropper may intercept any communication, she cannot "impersonate" Alice or Bob by sending fake messages on the channel. You can easily imagine the catastrophic consequences that such an attack could have: for example Eve could lie to Bob about Alice's choice of the test set T, making it a bit bigger; then Bob would reveal his outcomes in the bigger set, and Eve could keep them as additional side information about Alice's raw key.

This assumption is usually considered "benign", as indeed the access to an authenticated channel is a prerequisite for a large variety of cryptographic tasks. Thus in practice one imagines that any two parties Alice and Bob wanting to implement QKD have access to such a channel, that has been implemented by other means.

It is still interesting to consider how reasonable the assumption is, and how an authentication channel can be constructed. There are two main methods to achieve authentication, and each has its drawbacks. The first is to use public-key cryptography, which requires computational assumptions on the power of the eavesdropper. The second is to use private-key cryptography, which requires Alice and Bob to share a secret key... precisely the task QKD is meant to solve in the first place!

6.3.1 Everlasting security

The first method for authentication, based on public-key cryptography, can be quite efficient. We won't give any details here; the main primitive used is called a "digital signature", and it can be implemented based on any trapdoor one-way function (such as RSA, but of course with quantum adversaries you wouldn't want to rely on such an assumption!). You can check Chapter 5 in the notes [PS10] for more details.

But is it reasonable to make computational assumptions, when one of the main goals of quantum key distribution is to provide information-theoretic security? An argument in favor of this solution puts forward the property of "everlasting security". For the key generated in the protocol to be secure, it is sufficient that the CAC remains authenticated for the duration of the protocol, a few seconds at most. During this time it is crucial that Eve is not able to send fake messages. But once the protocol has ended, Alice and Bob have generated their private key, and it is no longer relevant whether the channel remains authenticated or not. So the computational assumption guaranteeing security of the authenticated channel only needs to hold for a few seconds, and the key generated in the protocol will remain forever secure: Eve has no information about it, and will not be able to gain any additional information by breaking a communication channel that is no longer in use!

6.3.2 Message authentication codes

The second method to achieve authentication is based on private-key cryptography. Let us consider a method to do this: even though it will not provide a good solution in practice (as using it to implement the CAC in QKD would require an initial shared secret key longer than the key generated by the protocol), it will still give you a good idea for the flavor of such constructions.

The main primitive used to achieve authentication is called a *message authentication code*, or MAC. A MAC is specified by a triplet of procedures:

- $Gen : () \to \mathscr{K}$ is the key generation procedure. It takes as input a security parameter n and returns a key $k \in \mathscr{K}$.
- $Tag : \mathscr{K} \times \mathscr{M} \to \mathscr{T}$ is the tagging procedure. It takes as input a key k and a message m, and returns a tag $\sigma = Tag_k(m)$.
- $Ver : \mathscr{K} \times \mathscr{M} \times \mathscr{T} \to \{0,1\}$ is the verification procedure. It takes as input a key, a message, and a (claimed) tag for the message. It returns either "1", meaning the tag is declared valid, or "0", meaning it is declared invalid.

All three procedures are required to run in polynomial time. The key generation procedure needs to be executed jointly by Alice and Bob, so that they have access to the same shared secret key k. Once this has been performed, they can separate. When Alice wants to send a message m to Bob (such as her choice of bases in the BB'84 protocol), she sends m accompanied with its tag $\sigma = Tag_k(m)$. Upon receiving (m,σ) Bob checks the tag by running the verification procedure $Ver_k(m,\sigma)$.

There are two requirements for a MAC. The first is correctness: it should always be the case that $Ver_k(m, Tag_k(m)) = 1$. The second is security. Informally, security of a MAC means that no adversary, given access to as many valid (message,tag) pairs as desired, is able to generate a message that it has not yet seen together with a valid tag for that message (except with probability negligible in the security parameter n).

Let's see a simple construction for a *one-time* MAC. A one-time MAC is a MAC that allows to tag a single message, but no more (i.e. security breaks down as soon as the adversary gets to see more than one valid (message,tag) pair generated using the same key). The construction is based on a family of two-universal hash functions. In week 4 we saw a construction of such a family $\mathscr{F} = \{f_y : \{0,1\}^n \to \{0,1\}^\ell\}$, for any $\ell \leq n$, that had size 2^{2n}. Given any such family,

- Gen returns a uniformly random $y \in \{1,\ldots,|\mathscr{F}|\}$ used to index a function f_y from $\mathscr{F}$, where ℓ is chosen as $\ell = n$.
- $Tag_y(m)$ returns $\sigma = f_y(m)$.
- $Ver_y(m,\sigma)$ returns 1 if and only if $\sigma = f_y(m)$.

This procedure is clearly correct. Why does it satisfy one-time security? Suppose given a valid (m,σ) pair. Using the property of two-universality, we know that for any $m' \neq m$ and for any σ', $\Pr_y(f_y(m) = \sigma \wedge f_y(m') = \sigma') = 2^{-2\ell}$. Given that the probability, over a random y, that $f_y(m) = \sigma$ is $2^{-\ell}$, applying Bayes' rule we get that

$$\Pr_y\left(f_y(m') = \sigma' \middle| f_y(m) = \sigma\right) = 2^{-\ell}.$$

In other words, the equation $f_y(m) = \sigma$ that the eavesdropper obtains on y when given a valid (m,σ) pair doesn't allow it to guess a valid σ', for any $m' \neq m$ of its choice, with probability more than $2^{-\ell}$, which is what a random guess would provide. Choosing $\ell \approx n$ thus gives us a construction of a one-time secure MAC.

Unfortunately, this MAC lets us authenticate messages of length n, provided we have a key of size $\log|\mathscr{F}| = 2n$. Not so useful! In general this is unavoidable, as any MAC with information-theoretic security requires a key as long as the total length of message that is to be tagged. For longer messages, one can again rely on computational assumptions (but weaker than for public-key

schemes: one-way functions are enough) to construct "many-times" MAC that allow to tag many messages with the same key. Or one can maintain information-theoretic security, but use more complicated methods for authentication that involve a multiple-round interaction between Alice and Bob. If you're interested in learning more, a good place to start is Chapter 5 in [PS10].

Acknowledgments

 The lecture notes are written by Nelly Ng, Thomas Vidick and Stephanie Wehner. We thank David Elkouss, Kenneth Goodenough, Jonas Helsen, Jérémy Ribeiro, and Jalex Stark for proofreading.

Bibliography

[BB84] C. H. Bennett and G. Brassard. “Quantum Cryptography: Public Key Distribution and Coin Tossing”. In: *Proceedings of IEEE International Conference on Computers, Systems and Signal Processing* (1984), pages 175–179 (cited on page 3).

[Eke91] Artur K Ekert. “Quantum cryptography based on Bell’s theorem”. In: *Physical review letters* 67.6 (1991), page 661 (cited on page 6).

[Pfi+16] Corsin Pfister et al. “Sifting attacks in finite-size quantum key distribution”. In: *New Journal of Physics* 18.5 (2016), page 053001 (cited on page 5).

[PS10] Rafael Pass and Abhi Shelat. *A Course in Cryptography*. Lecture notes available at `http://www.cs.cornell.edu/courses/cs4830/2010fa/lecnotes.pdf`. 2010 (cited on pages 10, 12).

[TL15] Marco Tomamichel and Anthony Leverrier. “A rigorous and complete proof of finite key security of quantum key distribution”. In: *arXiv preprint arXiv:1506.08458* (2015) (cited on page 9).

[Wie83] Stephen Wiesner. “Conjugate Coding”. In: *SIGACT News* 15 (1983), pages 78–88 (cited on page 3).

Lecture Notes

Quantum Cryptography Week 7:

Quantum cryptography using untrusted devices

Contents

This week we introduce a new variant of the BB'84 quantum key distribution protocol we studied last week. This variant is due to Ekert [Eke91] and is often referred to as the E'91 protocol for quantum key distribution (since our protocol won't exactly follow Ekert's original proposal we will simply call it the "DIQKD protocol". Although it looks rather similar to the BB'84 protocol, or more specifically its purified version, the key difference is that we use a different test for step 7. in the protocol (see the description of BB'84 from last week's notes).

Instead of the "matching outputs" test considered in the BB'84 protocol, Ekert's protocol uses a test based on the CHSH game (recall the game from week 2!). The advantage of using this test is that it allows us to prove that the protocol is secure without relying on Alice and Bob performing trusted measurements on their qubit in each round — in fact, without even relying on the assumption that the system they measure is a qubit! This stronger notion of security is called *device-independent security*, and we'll define it in more detail later in these notes.

Before introducing Ekert's protocol and its analysis, we first return to the CHSH game and give a more detailed analysis of its properties than we did in week 2. This game turns out to have a striking property, which forms the key to its use in the DIQKD protocol. This is the property of *rigidity*, which states that optimal strategies for the players in the game are unique in a very strong sense: any strategy that achieves the optimum success probability $p^*_{\text{CHSH}} = \cos^2 \pi/8$, or even close to the optimum, must be equivalent (in a sense to be made precise later) to the strategy we saw in week 2. There is no alternative! As an immediate consequence we get that any strategy with close to optimal success probability must involve a shared entangled state between Alice and Bob that is equivalent to an EPR pair, just as the optimal strategy we described does. This fact does not need us to assume any a priori knowledge about the state or the measurements used in the strategy.

7.1 Testing EPR pairs

Recall that in the CHSH game the referee sends each of the two players, Alice and Bob, a uniformly random bit $x, y \in \{0,1\}$ respectively. The players have to return outcomes $a, b \in \{0,1\}$ such that the CHSH condition $a \oplus b = x \wedge y$ is satisfied. We saw that the maximum success probability of classical non-communicating players in this game is $p_{\text{CHSH}} = 3/4$, while if Alice and Bob are quantum there is a strategy that allows them to succeed with probability $p^*_{\text{CHSH}} = \cos^2 \pi/8 \approx 0.85$.

In the strategy we described, Alice and Bob share an EPR pair $|\phi^+\rangle_{AB}$ and make the following measurements. When $x = 0$, Alice measures her qubit in the standard basis $\{|0\rangle, |1\rangle\}$, and when $x = 1$ she measures in the Hadamard basis $\{|+\rangle, |-\rangle\}$. When $y = 0$, Bob measures his qubit in the basis $\{\cos(\pi/8)|0\rangle + \sin(\pi/8)|1\rangle, -\sin(\pi/8)|0\rangle + \cos(\pi/8)|1\rangle\}$ and when $y = 1$, he measures in the basis $\{\cos(\pi/8)|0\rangle - \sin(\pi/8)|1\rangle, \sin(\pi/8)|0\rangle + \cos(\pi/8)|1\rangle\}$. Since these measurements are binary projective measurements, with POVM elements of the form $\{\Pi, \mathbb{I} - \Pi\}$, we can equivalently describe them using the associated *observables* $O = \mathbb{I} - 2\Pi$. Note that O is a Hermitian operator which squares to identity. For Alice's measurements the observables are

$$A_0 = |0\rangle\langle 0| - |1\rangle\langle 1| = Z \ (x = 0) \quad \text{and} \quad A_1 = |+\rangle\langle +| - |-\rangle\langle -| = X \ (x = 1).$$

For Bob we have

$$B_0 = H \ (y = 0) \quad \text{and} \quad B_1 = \tilde{H} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 \\ -1 & -1 \end{pmatrix} \ (y = 1).$$

We introduced this as a "good" strategy for the players: it certainly beats the classical bound $p_{\text{CHSH}} = 3/4$, and achieves $p^*_{\text{CHSH}} = \cos^2 \pi/8$. But could there be better strategies, achieving an even larger value? Or, even if they are not better, different strategies, based on using a different type of entangled state, for achieving the same success probability?

We're going to show that this is not the case: the maximum success probability of any quantum strategy in the CHSH game, as complicated as it may be, is p^*_{CHSH}. Moreover, any strategy

achieving this value must be "equivalent" to the strategy described above. What do we mean by equivalent? We couldn't possibly hope to claim that the strategy is strictly unique. For example, if Alice and Bob were to rotate their basis choices by the same angle, then since the EPR pair is itself rotation invariant their success probability would remain unchanged. The theorem shows that this local degree of freedom is essentially the only flexibility that the players have in designing an optimal strategy.

Theorem 7.1.1 — CHSH rigidity. Suppose given an entangled state $|\psi\rangle_{AB} \in \mathbb{C}^{d_A} \otimes \mathbb{C}^{d_B}$ and observables A_0, A_1 for Alice and B_0, B_1 for Bob such that the corresponding strategy has a success probability $p^*_{\text{CHSH}} = \cos^2 \pi/8$ in the CHSH game. Then there exist local isometries $U_A : \mathbb{C}^{d_A} \to \mathbb{C}^2 \otimes \mathbb{C}^{d_{A'}}$ and $V_B : \mathbb{C}^{d_B} \to \mathbb{C}^2 \otimes \mathbb{C}^{d_{B'}}$ such that

$$U_A \otimes V_B |\psi\rangle_{AB} = |\phi^+\rangle \otimes |\text{junk}\rangle_{\text{A'B'}}$$

and

$$\begin{aligned}
(U_A \otimes V_B)(A_0 \otimes \mathbb{I}_B)|\psi\rangle &= ((Z \otimes \mathbb{I})|\phi^+\rangle) \otimes |\text{junk}\rangle, \\
(U_A \otimes V_B)(A_1 \otimes \mathbb{I}_B)|\psi\rangle &= ((X \otimes \mathbb{I})|\phi^+\rangle) \otimes |\text{junk}\rangle, \\
(U_A \otimes V_B)(\mathbb{I}_A \otimes B_0)|\psi\rangle &= ((\mathbb{I} \otimes H)|\phi^+\rangle) \otimes |\text{junk}\rangle, \\
(U_A \otimes V_B)(\mathbb{I}_A \otimes B_1)|\psi\rangle &= ((\mathbb{I} \otimes \tilde{H})|\phi^+\rangle) \otimes |\text{junk}\rangle.
\end{aligned}$$

In words, the theorem says that if a strategy achieves the optimal value in CHSH then up to some local rotations on Alice and Bob's spaces it looks exactly as the strategy described above. We called the rotation "isometries" because their range might not be the whole space; in particular it is not necessarily the case that d_A or d_B are even. The state $|\text{junk}\rangle$ is an arbitrary state that does not matter for the purposes of analyzing the strategy. This state is unavoidable, as any strategy can always be made to appear more complicated by extending the entangled state arbitrarily, and making the players' measurements act as identity on the extended space.

Note also the theorem presupposes that the players' strategy can be described by observables, or equivalently binary projective measurements. More generally we may consider players that apply a non-projective POVM. However, a POVM can always be simulated with a projective measurement acting on a larger space, so the assumption is without loss of generality.

R In practice it will never be possible to certify that a given device implements a strategy with optimal success probability in the CHSH game: at best, by repeated testing it will be possible to verify that it achieves a success probability at least $p^*_{\text{CHSH}} - \delta$, where $\delta > 0$ is a quantity depending on the quality of the device and on the accuracy of the testing performed (i.e. the number of repetitions of the game). To handle this it is important to obtain "robust" analogues of Theorem 7.1.1. Such a result is known, where the exact equalities in Theorem 7.1.1 are replaced by approximations in trace distance with an error scaling as $O(\sqrt{\delta})$ [**mckague2012robust**].

Before we get to the proof of the theorem we make a small detour and explore the notion of angle between a pair of projection operators.

7.1.1 Principal angles and Jordan's lemma

Consider two lines through the origin in the complex plane $\mathbb{C}^2$. Each line is described by a unit vector $|u\rangle$, $|v\rangle$, and (ignoring any orientation) the angle between the two lines is the unique $\theta \in [0, \pi/2)$ such that $\cos^2 \theta = |\langle u|v\rangle|^2$. Up to a change of basis we can always consider that $|u\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ and (up to an irrelevant phase) $|v\rangle = \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix}$. A more pedantic way to describe the

angle between the two lines is through the associated rank-1 projections $P = |u\rangle\langle u|$ and $Q = |v\rangle\langle v|$: there will always exist a choice of basis for $\mathbb{C}^2$ in which

$$P = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \qquad \text{and} \qquad Q = \begin{pmatrix} \cos^2\theta & \cos\theta\sin\theta \\ \cos\theta\sin\theta & \sin^2\theta \end{pmatrix},$$

for some $\theta \in [0, \pi/2)$.

How do we generalize the notion of angle to higher dimensional subspaces? The notion of principal angle provides an inductive definition. Suppose P and Q are two orthogonal projections in $\mathbb{C}^d$. (We identify the projections with the space on which they project.) The smallest principal angle between P and Q is defined as $\theta_1 \in [0, \pi/2)$ such that

$$\cos^2\theta_1 = \sup_{|u\rangle\in P, |v\rangle\in Q} |\langle u|v\rangle|^2,$$

where by $|u\rangle \in P$ we mean any unit vector in the range of P, i.e. such that $P|u\rangle = |u\rangle$. This is a natural definition: we are finding the lines lying in P and Q that form the smallest possible angle . If P and Q intersect, then they share a vector and $\theta_1 = 0$.

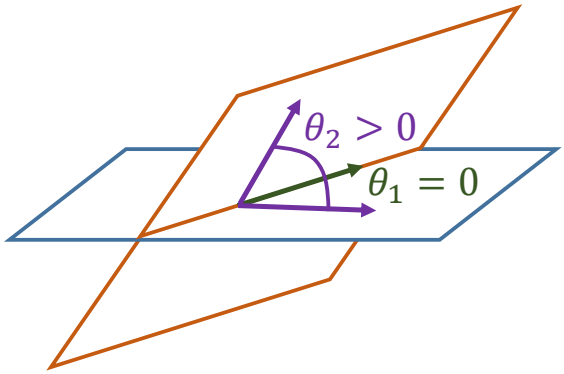

Figure 7.1: Principal angles between two 2-dimensional subspaces in 3 dimensions. The subspaces intersect, and the smallest angle is $\theta_1 = 0$. The second principal angle is $\theta_2 > 0$.

We define principal angles $\theta_2, \ldots, \theta_d$, where $d = \min(\operatorname{rank} P, \operatorname{rank} Q)$, inductively via

$$\cos^2\theta_i = \sup_{\substack{|u_i\rangle\in P,\, |u_i\rangle\perp \mathrm{Span}\{|u_1\rangle,\ldots,|u_{i-1}\rangle\} \\ |v_i\rangle\in Q,\, |v_i\rangle\perp \mathrm{Span}\{|v_1\rangle,\ldots,|v_{i-1}\rangle\}}} |\langle u_i|v_i\rangle|^2,$$

where $|u_1\rangle, \ldots, |u_{i-1}\rangle$ are unit vectors in P that achieve the optimum in the definition of $\theta_1, \ldots, \theta_{i-1}$ respectively, and similarly for the $|v_j\rangle$ and Q.

Jordan's lemma states that associated with the principal angles comes a very convenient simultaneous block decomposition of P and Q.

Lemma 1 — Jordan's lemma. Let P and Q be two projection operators in $\mathbb{C}^d$. Then there exists a basis of $\mathbb{C}^d$ in which P and Q are simultaneously block diagonal, with blocks of size one or two such that either (for one-dimensional blocks)

$$P, Q \in \{(0), (1)\},$$

or (for two-dimensional blocks)

$$P = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \qquad Q = \begin{pmatrix} \cos\theta_i^2 & \cos\theta_i\sin\theta_i \\ \cos\theta_i\sin\theta_i & \sin\theta_i^2 \end{pmatrix},$$

with $\theta_1, \ldots, \theta_d \in (0, \pi/2]$, $d = \min(\operatorname{rank} P, \operatorname{rank} Q)$, the principal angles between P and Q.

The proof of the lemma is not very hard, and considers an alternate definition of the principal angles via the singular values of the operator PQ (see e.g. Exercise VII.1.10 of [Bha13]).

7.1.2 Proof of the rigidity theorem

The proof of Theorem 7.1.1 proceeds in two steps. In the first step we use Jordan's lemma to reduce the case of general strategies to the case of "qubit strategies", for which the shared state is a two-qubit entangled states and the players' observables single-qubit observables. In the second step we analyze qubit strategies in detail and show that they must take the form of Pauli measurements on an EPR pair.

1. Reduction to qubit strategies.

Consider an arbitrary strategy $|\psi\rangle_{AB}$, A_0, A_1, B_0, B_1. Apply Jordan's lemma to the projections $P = \frac{1}{2}(\mathbb{I} + A_0)$ and $Q = \frac{1}{2}(\mathbb{I} + A_1)$. The lemma gives a basis for Alice's space $\mathbb{C}^{d_A}$ such that both P and Q are block-diagonal in that basis, with blocks of size at most 2×2. Then $A_0 = 2P - \mathbb{I}$ and $A_1 = 2Q - \mathbb{I}$ are block-diagonal in the same basis.

This block-diagonal decomposition lets us reformulate Alice's strategy as follows: each of her two-outcome projective measurements is equivalent to a measurement which (i) applies a multiple-outcome projective measurement that projects on the individual blocks of the decomposition, and (ii) depending on the block obtained as outcome performs the basis measurement associated with the restriction of A_0 (or A_1) to that block.

Exercise 7.1.1 Suppose that after application of Jordan's lemma we discover a basis

$$\{|u_1\rangle, |u_2\rangle, |u_3\rangle, |u_4\rangle, |u_5\rangle\} \tag{7.1}$$

of $\mathbb{C}^5$ in which

$$A_0 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad A_1 = \begin{pmatrix} \frac{1}{2} & -\frac{1}{2} & 0 & 0 & 0 \\ -\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 \end{pmatrix}.$$

Consider the two-outcome projective measurements associated with A_0 and A_1. Give an equivalent description of these measurements as the combination of a projective measurement $\{\Pi_0, \Pi_1, \Pi_2\}$ followed by a basis measurement involving at most 2 basis elements. The projective measurement should be independent of Alice's input x, while the basis measurement should depend both on the outcome of the projective measurement and Alice's input. ∎

The same argument can be applied to Bob's observables. Now the key point is that, since the block decomposition is the same for A_0 and A_1 (resp. B_0 and B_1), step (i) associated with projection on the blocks does not depend on the player's question. Thus the step could be performed even before the game even starts, without affecting their success probability! But then the players are really playing the game with a qubit strategy — whichever qubit strategy corresponds to the outcomes they obtained when applying the projective measurement from step (i).

This reformulation of an arbitrary strategy shows that it can always be reduced to a convex combination of qubit strategies, and it will be sufficient to analyze the latter.

2. Optimal qubit strategies.

To prove the theorem we first express the success probability p^*_{win} of a given quantum strategy in terms of the observables A_x and B_y.

Exercise 7.1.2 Using the definition of the winning criterion $a \oplus b = x \wedge y$ and the relation between observables and binary measurements, show that

$$p^*_{\text{win}} = \frac{1}{2} + \frac{1}{8}\langle\psi|A_0 \otimes B_0 + A_0 \otimes B_1 + A_1 \otimes B_0 - A_1 \otimes B_1|\psi\rangle. \tag{7.2}$$

■

Let's call the operator appearing inside the bra-ket in (7.2) the CHSH operator,

$$\text{CHSH} = \text{A}_0 \otimes \text{B}_0 + \text{A}_0 \otimes \text{B}_1 + \text{A}_1 \otimes \text{B}_0 - \text{A}_1 \otimes \text{B}_1.$$

The main trick in the proof is to consider the square of this operator. Using $A_0^2 = A_1^2 = B_0^2 = B_1^2 = \mathbb{I}$, we get

$$\begin{aligned}
\text{CHSH}^2 &= \big((A_0 + A_1) \otimes B_0 + (A_0 - A_1) \otimes B_1\big)^2 \\
&= (A_0 + A_1)^2 \otimes \mathbb{I} + (A_0 - A_1)^2 \otimes \mathbb{I} + (A_0 + A_1)(A_0 - A_1) \otimes B_0 B_1 \\
&\quad + (A_0 - A_1)(A_0 + A_1) \otimes B_1 B_0 \\
&= 4\mathbb{I} + [A_0, A_1] \otimes [B_1, B_0],
\end{aligned} \tag{7.3}$$

where $[A_0, A_1] = A_0A_1 - A_1A_0$ and $[B_1, B_0] = B_1B_0 - B_0B_1$ are the commutators. Since the operator norm (the largest singular value) of $[A_0, A_1]$ and $[B_0, B_1]$ is each at most 2, the norm of CHSH^2 (which is simply the largest overlap of CHSH^2 with a unit vector) is at most 8. Plugging back into (7.2), even an optimal choice of $|\psi\rangle$ (i.e. an eigenvector of CHSH associated to its largest singular value) will give a value at most $p^*_{\text{win}} \leq 1/2 + \sqrt{8}/8 = \cos^2 \pi/8 = p^*_{\text{CHSH}}$. Thus $\cos^2 \pi/8$ is indeed the maximum probability of success in the CHSH game.

Note that so far we have not used the reduction to qubit strategies discussed in the previous section, and the preceding argument is completely general. Let's now assume we are working with a qubit strategy which achieves the optimal $p^*_{\text{win}} = p^*_{\text{CHSH}}$. Then all inequalities discussed above must be tight. In particular, $|\psi\rangle$ must be an eigenvector of CHSH with eigenvalue $2\sqrt{2}$, and as a consequence of (7.3) $|\psi\rangle$ must also be an eigenvector of $[A_0, A_1] \otimes [B_0, B_1]$ with associated eigenvalue 4. Squaring this operator,

$$\big([A_0, A_1]^2 \otimes [B_0, B_1]^2\big)|\psi\rangle = 16|\psi\rangle.$$

Using further that $[A_0, A_1]^2 \leq 4\mathbb{I}$ and $[B_0, B_1]^2 \leq 4\mathbb{I}$ we get that necessarily

$$\big([A_0, A_1]^2 \otimes \mathbb{I}\big)|\psi\rangle = \big(\mathbb{I} \otimes [B_0, B_1]^2\big)|\psi\rangle = 4|\psi\rangle, \tag{7.4}$$

as neither operator can reduce the norm of $|\psi\rangle$. Assume $|\psi\rangle$ is not trivial, in the sense that its reduced density matrices on A and B have rank 2 (if this is not the case then it is easy to see that the strategy boils down to a classical strategy, which cannot achieve a success probability larger than $p_{\text{CHSH}} = 3/4$). Tracing out the A or B qubits in (7.4) and inverting the reduced density matrix of $|\psi\rangle$ on the remaining qubit gives us the operator equalities $A_0A_1 = -A_1A_0$ and $B_1B_0 = -B_0B_1$: Alice's and Bob's observables pairwise anti-commute. It turns out that anti-commutation is a surprisingly strong constraint, as shown in the following exercise.

Exercise 7.1.3 Suppose that R and S are two observables on $\mathbb{C}^2$ such that $RS = -SR$. Then there exists a basis of $\mathbb{C}^2$ in which $R = Z$ and $S = X$. [Hint: first show that we cannot have $R = \mathbb{I}$ or $R = -\mathbb{I}$, and deduce the eigenvalues of R. Use this to write R in a convenient form, and then use the anti-commutation relation to express S.] ■

Applying the results of the exercise to A_0 and A_1 we obtain a rotation U_A on Alice's qubit such that $U_A A_0 U_A^\dagger = Z$ and $U_A A_1 U_A^\dagger = X$. Similarly, for Bob's observables we may find a unitary U_B such that $U_B B_0 U_B^\dagger = H$ and $U_B B_1 U_B^\dagger = \tilde{H}$. Note that for Bob we are using H and $\tilde{H}$ in lieu of X and Z, but any pair of single-qubit observables will do. To conclude it remains to show the following.

Exercise 7.1.4 Show that the operator

$$Z \otimes H + X \otimes H + X \otimes \tilde{H} - Z \otimes \tilde{H}$$

has largest eigenvalue $2\sqrt{2}$, with a unique associated eigenvector equal to $|\phi^+\rangle$. ■

3. Putting everything together.

We are almost done with the proof of Theorem 7.1.1. To summarize, we start with an arbitrary strategy $|\psi\rangle_{AB}$, A_0, A_1, B_0, B_1 with success probability $p^*_{\text{win}} = p^*_{\text{CHSH}}$ in the CHSH game. Using part 1. this strategy can be decomposed in a convex combination of qubit strategies. More formally, there are projective measurements $\Pi^A = \{\Pi^A_1, \ldots, \Pi^A_{k_A}\}$ and $\Pi^B = \{\Pi^B_1, \ldots, \Pi^N_{k_B}\}$ for Alice and Bob, made of projectors with rank at most 2 each, such that $A_x = \sum_j \Pi^A_j A_x \Pi^A_j$ and $B_y = \sum_j \Pi^B_j B_y \Pi^B_j$. The associated block decomposition can be specified by a unitary changes of basis U'_A and U'_B on Alice and Bob's systems respectively.

Using the first steps of part 2., we know that any strategy can have success probability at most p^*_{CHSH}, therefore all the qubit strategies, given by $(\Pi^A_j \otimes \Pi^B_\ell |\psi\rangle, \Pi^A_j A_x \Pi^A_j, \Pi^B_\ell B_y \Pi^B_\ell)$ for any $j \in \{1, \ldots, k_A\}$ and $\ell \in \{1, \ldots, k_B\}$, must have success probability p^*_{CHSH} (otherwise the overall strategy wouldn't achieve the optimal success probability).

By the remainder of part 2., for of these qubit strategies there exists a local change of basis U^A_j and U^B_ℓ in which it is equivalent to the canonical optimal strategy. By combining the unitaries U_A (resp. U_B), which specify the blocks, with the unitaries U^A_j (resp. U^B_ℓ), which identify a basis for each block in which $\Pi^A_j A_0 \Pi^A_j = Z$, $\Pi^A_j A_1 \Pi^A_j = X$, and similarly for Bob and $H, \tilde{H}$, we obtain the isometries claimed in the theorem: the proof is complete!

7.2 A device independent QKD protocol

In the previous section we gave mathematical justification for our intuition that the CHSH game can serve as a good test for entanglement. We're now going to see how the game can be embedded as a test in a key distribution protocol to make the protocol *device-independent*. Let's first explore more precisely what this notion of security covers — and does not cover.

7.2.1 Device-independent security

Device independence is a notion of security for cryptography that is motivated by the practical difficulty of characterizing the quantum mechanical devices, such as photon emitters or receptors, used in protocols such as the one for BB'84. The protocol calls for Alice to "prepare a qubit in the Hadamard basis", and for Bob to "measure his qubit in the $\pi/8$-rotated basis". When Alice prepares her qubit, and when Bob measures it, can they really trust their equipment to implement the prescribed task? What if Bob's measurement apparatus fails some percentage of the time: should he treat these failures as noise, or could they be adversarial (for example, the failure rate could vary depending on his basis choice or on the state of the qubit)? What if Alice's preparation device sometimes created two qubits, instead of a single one, without her noticing; could the additional qubit be intercepted by Eve and provide her with additional information, without Alice or Bob noticing? The following exercise shows that these such misbehavior of Alice and Bob's equipment can lead to serious security issues.

■ **Example 7.2.1 — Taken from (Pir+09).** Consider the purified variant of the BB'84 protocol. Suppose that Eve prepares a state ρ_{ABE} of the following form:

$$\rho_{ABE} = \sum_{x,z=0}^{1} |x,z\rangle\langle x,z|_A \otimes |x,z\rangle\langle x,z|_B \otimes |x,z\rangle\langle x,z|_E. \tag{7.5}$$

Now suppose Alice and Bob's measurement devices, instead of measuring a single qubit in the standard or Hadamard bases, as they think the device does, in fact performs the following:

- When the device is told to measure in the standard basis, it measures the first qubit of the two-qubit system associated with the device in (7.5) in the standard basis;
- When the device is told to measure in the Hadamard basis, it measures the second qubit of the two-qubit system associated with the device in (7.5) in the standard basis.

If the devices perform as described they perfectly pass all tests performed in the protocol: indeed, when the basis choice is the same the outcome is the same, whereas when the bases are different the outcomes are perfectly uncorrelated. But any key extracted from ρ_{ABE} in (7.5) is completely insecure! (Exercise: Give an explicit attack for Eve.) ■

The difficulty is not only theoretical. In fact, one of the first "attacks" on the BB'84 protocol is that the photon receptor used in an early experiment made a different clicking noise when it measured in one of Bob's bases, thereby "leaking" Bob's basis choice to any eavesdropper within earshot! Many such side-channel attacks have been demonstrated, and implemented, in practice. Some of the most effective are called "detector blinding" attacks, in which the eavesdropper can take complete control of Bob's receptor by shining a very bright laser right into it (without Bob noticing!).

Device-independence aims to guarantee security even in the context of such seemingly dramatic failures of Alice and Bob's equipment. But we have to be careful what we promise exactly. For example, at the extreme we could imagine that Bob's device contains radio equipment that automatically transmits all its measurement results to Eve: in this case security is compromised, but there is no way for Bob to detect the radio transmitter unless he opens the device. In a similar vein, if the random number generator used by Alice to make her basis choices is biased, or controlled by Eve, then security cannot hold. The specific kinds of failures that are allowed by a device-independent proof of security thus have to be specified on a case-by-case basis. For quantum key distribution we will make the following assumptions:

1. Alice and Bob's labs are perfectly isolated: once the protocol starts no information enters or exits their respective labs that is not specified in the protocol.
2. Alice and Bob's random number generators are perfect.
3. The devices used by Alice and Bob to perform measurements are arbitrary. These devices are initialized in a state ρ_{ABE} that may be chosen by the adversary. At each step of the protocol, each of Alice and Bob's devices makes a measurement when instructed, and always produces an outcome $x \in \{0,1\}$. The measurement that is performed is arbitrary. In particular the device may have memory and behave differently in each round.
4. At the end of the protocol the devices are discarded and will never be re-used. It is assumed that they will never fall in Eve's hands.

Device-independence refers to the freedom in assumption 3., which allows the devices to perform any kind of measurement, on any state; both may have been decided on by Eve as part of her "attack".

The last assumption is important: as will be apparent from the protocol, the devices themselves know what Alice and Bob's raw key is, and could potentially store it in memory. It is important that this memory is never allowed to leak to any adversary.

7.2.2 The protocol

The security of the DIQKD protocol we are about to give, a variant of Ekert's original proposal for quantum key distribution [Eke91], is based on the rigidity properties of the CHSH game that we explored earlier on: a high success probability in the game can be used to certify an EPR pair, even when the measurements being performed could a priori be arbitrary.

Before we proceed, there is a small technicality we have to deal with. In the honest optimal strategy for the CHSH game it is never the case that Alice and Bob use the same basis, and thus they never produce perfectly correlated outcomes. In order to produce a key it will be convenient for them to be able to rely on (almost) perfectly correlated outcomes for at least one choice of a pair of inputs. Therefore in the protocol we think of Bob's device as having 3, instead of 2, possible inputs: the inputs $\tilde{\theta} \in \{0,1\}$ correspond to the usual CHSH inputs (for which the ideal device would measure using observables H and $\tilde{H}$ respectively), and the additional input $\tilde{\theta} = 2$ instructs the device to measure in the standard basis, so that on inputs $(\theta, \tilde{\theta}) = (0,2)$ the devices are expected to produce matching outcomes (of course, in practice the device may implement any measurement it likes).

Protocol 1 Device independent QKD. Outputs $k \in \{0,1\}^\ell$ to both Alice and Bob.

1. Alice chooses a uniformly random basis string $\theta = \theta_1, \ldots, \theta_n \in \{0,1\}^n$ and sequentially instructs her measurement device to measure in the bases θ. The device returns a string of outcomes $x = x_1, \ldots, x_n$.
2. Bob chooses a uniformly random basis string $\tilde{\theta} = \tilde{\theta}_1, \ldots, \tilde{\theta}_n \in \{0,1,2\}^n$ and sequentially instructs his measurement device to measure in the bases $\tilde{\theta}$. The device returns a string of outcomes $\tilde{x} = \tilde{x}_1, \ldots, \tilde{x}_n$.
3. Alice and Bob tell each other their basis strings θ and $\tilde{\theta}$ respectively over the CAC.
4. Alice selects a random subset $T \subseteq [n]$ of size $n/2$ and announces T to Bob. They set $T' = \{j \in T, \tilde{\theta}_j \in \{0,1\}\}$, $T'' = \{j \in T, \theta_j = 0 \wedge \tilde{\theta}_j = 2\}$, and $R = \{j \notin T, \theta_j = 0 \wedge \tilde{\theta}_j = 2\}$.
5. Alice and Bob announce x_T and $\tilde{x}_T$ to each other over the CAC. They compute the success probabilities $p_{\text{win}} = |\{j \in T', x_j \oplus \tilde{x}_j = \theta_j \wedge \tilde{\theta}_j\}|/|T'|$ and $p_{\text{match}} = |\{j \in T'', x_j = \tilde{x}_j\}|/|T''|$. If $p_{\text{win}} < \cos^2 \pi/8 - \delta$ or $p_{\text{match}} < 1 - \delta$ they abort.
6. Alice and Bob perform information reconciliation and privacy amplification on their respective outcomes $x_R, \tilde{x}_R$.

As already mentioned security of the protocol is based on the rigidity of the CHSH correlations. However, as we already saw in last week's analysis, the kind of strong guarantees provided by Theorem 7.1.1 are very difficult to expand to the analysis of a full protocol, where not just one but many CHSH games are played sequentially. Luckily, these guarantees are also more than we really need: ultimately, what need to show is security of the classical key — however it is obtained at the quantum mechanical level. In fact, due to the last steps of information reconciliation and privacy amplification (which are unchanged from the BB'84 protocol) the only thing we really need to establish is uncertainty in Alice's outputs x_R, given the side information E. To show this, we use yet another variant of the guessing game, this time based on the CHSH correlations.

7.2.3 A CHSH-based guessing game

Consider the following guessing game. There are three players, Alice, Bob and Eve. Alice receives an input $\theta \in \{0,1\}$, Bob receives a $\tilde{\theta} \in \{0,1,2\}$, and Eve receives no input (equivalently, her input is always the same). The players produce outcomes $x, \tilde{x}, z \in \{0,1\}$ respectively. They win the game if and only if the following conditions hold:

- If $\tilde{\theta} \in \{0,1\}$ then $x \oplus \tilde{x} = \theta \wedge \tilde{\theta}$.
- If $\theta = 0$ and $\tilde{\theta} = 2$ then $x = z$.

Lemma 2 — CHSH guessing lemma. Consider an arbitrary strategy for the players in the CHSH

guessing game. Let p_{win} be the probability that the first test passes (conditioned on $\tilde{\theta} \in \{0,1\}$) and p_{id} the probability that the second test passes (conditioned on $\theta = 0$ and $\tilde{\theta} = 2$). Suppose that $p_{\text{win}} \geq \cos^2 \pi/8 - \delta$. Then $p_{\text{id}} \leq 1/2 + 2\delta^{1/2}$.

We will not give a proof of the lemma here. There are many ways it can be shown, yielding bounds of varying quality. The simplest analysis would consider a relaxation of the problem where the three players are allowed any kind of *non-signaling strategy*: in this case a bound can be obtained via linear programming. The bound can then be strengthened by considering the fact that the players must be quantum, using a semidefinite relaxation of the problem. But the optimal bound can be obtained by a direct analytic calculation, using the fact that Alice only has two possible inputs to reduce to the two-dimensional case via an application of Jordan's lemma. This is done in [Pir+10], from which the bound given here, which is due to [VV14], can be derived.

7.3 Security of device-independent quantum key distribution

Let's analyze the security of our DIQKD protocol. Our goal is to show that there is an $\varepsilon > 0$ (the error) and a $\kappa > 0$ (the key rate), depending on the parameters of the protocol, such that the following holds:

> For any strategy of the eavesdropper Eve, specified by an initial state ρ_{ABE} of the devices and a choice of measurements to be made at every step in the protocol, either Alice and Bob abort in step 5. of the protocol with probability larger than ε, or Alice's outcomes x_R at step 6. satisfy $\mathrm{H}_{\min}^{\varepsilon}(X_R|EK) \geq \kappa n$, where K denotes all the communication exchanged on the CAC during the protocol.

A few comments regarding this statement. First, by focusing on establishing a sufficiently large rate of min-entropy in Alice's raw key bits we are putting the steps of information reconciliation and privacy amplification behind us. We studied these steps in detail in previous weeks and understand them well, but they have to be performed, and as a result the length of key produced will be slightly reduced.

Second, ε enters in the statement twice. First, we are assuming that the probability of an abort in step 5. is not too large, not larger than $1 - \varepsilon$. The reason this is needed is that conditioning on very low probability events can have drastic consequences. There is always the chance that Eve prepares states that have a very high failure probability, but such that conditioned on passing all the tests (which might still happen with low probability — for instance in the extremely unlikely event that the sets T' and T'' are both empty!) the protocol becomes completely broken. Second, ε also appears in the bound $\mathrm{H}_{\min}^{\varepsilon}(X_R|EK) \geq \kappa n$. This bound is evaluated on the joint state of Alice and Eve in step 6., conditioned on not aborting in step 5. It is unrealistic to hope to prove a bound directly on the min-entropy of that state. For instance, even though Alice and Bob did not abort there is always a small chance that Eve still attacked a large number of rounds of the protocol (by preparing a malicious entangled state) but got extremely lucky in the tests. Thus we will only be able to show that the state at step 6. is close, in trace distance, to a state whose min-entropy is large; this is the meaning of the ε in the smooth min-entropy condition $\mathrm{H}_{\min}^{\varepsilon}(X_R|EK)$.

Now that we understand precisely our target — let's prove security! There are two main steps. The first is to use the testing condition from step 5. to infer a lower bound on the conditional min-entropy $\mathrm{H}_{\min}(X_j|EK)$ in individual rounds of the protocol, for $j \in R$. The second is to combine these bounds into a bound on the whole string X_R.

We will show how both steps can be performed under the restriction that the eavesdropper is limited to so-called *collective attacks*. A collective attack is one in which the initial state of the devices takes the form $\rho_{ABE}^{\otimes n}$, and moreover the measurement performed by the device in each round

is the same (on the same inputs), i.e. the device is memoryless. (The name "collective" comes from the fact that at the end of the protocol we still allow Eve to perform a joint measurement simultaneously on all the E systems, as well as all the classical information she has acquired, when making her best guess for the key.)

The most general attacks, without these two assumptions, are called *coherent attacks*. These allow Eve to introduce significant complications by choosing an initial state that is entangled across all rounds; in fact the state may not have n pre-specified qubits and the device could measure the same, or partially overlapping, high-dimensional systems in different rounds. This makes the analysis much more involved, and we will only outline an important tool that can be used to adapt our security proof against collective attacks to a full proof of security against coherent attacks.

7.3.1 Collective attacks

The assumption of collective attacks allows us to model the behavior of the device in each round as independent from its actions in previous (or subsequent) rounds. In particular, the device has a well-defined success probability in the CHSH game: if it is given inputs $\theta, \tilde{\theta} \in \{0,1\}$ in any particular round, how well does it perform in the game?

This is precisely the quantity that is estimated at step 5. of the protocol. Let $Z_1, \ldots, Z_k$, where $k = |T'|$, be binary random variables such that Z_j equals 1 if the CHSH condition in round j is satisfied. Then $p_{\text{win}} = |T'|^{-1} \sum_{j \in T'} Z_j$. Note that this is an "observed" quantity; let $\hat{p}_{\text{win}}$ be the "true value", i.e. the probability of success of the device in the CHSH game. How different can p_{CHSH} and $\hat{p}_{\text{CHSH}}$ be?

We can think of the inputs for the rounds T' as being selected after the set of rounds T' itself is chosen by Alice: for instance, we could imagine Bob choosing rounds in which $\tilde{\theta}_J = 2$ at random, and Alice choosing a random set T; this defines the set T' but the players still have the freedom to choose specific inputs for those rounds. Since the probability of any given round lying in T is $1/2$, and independently the probability that Bob chooses $\tilde{\theta}_j = 2$ is $1/3$, the expected size of $|T'|$ is $n/6$. To show that the chance that the actual size differs from the expected size by too much is small we need a simple concentration inequality.

Theorem 7.3.1 — Chernoff bound (Che81). Let $X_1, \ldots, X_n$ be i.i.d. random variables taking values in $\{0,1\}$, and $\mu = \mathrm{E}[X_i]$. Then for all $0 < \alpha < 1$,

$$\Pr\left(\left|\frac{1}{n}\sum_{i=1}^{n} X_i - \mu\right| > \alpha\mu\right) \leq 2e^{-\frac{\alpha^2 \mu n}{3}}.$$

If we apply the proposition with $\mu = 1/6$ and $\alpha = 1/4$ we obtain that the probability that $|T'| < n/8$ is at most $e^{-n/(3\cdot 6\cdot 16)}$. Let's assume this is not the case. Then we can apply the same bound once more to obtain

$$\Pr\left(\sum_{j \in T'} Z_j > (1+\alpha)|T'|\hat{p}_{\text{win}}\right) \leq 2e^{-\frac{\alpha^2 \hat{p}_{\text{win}}|T'|}{3}}.$$

Hence, using our lower bound on the size of $|T'|$ as well as $\hat{p}_{\text{win}} \geq 1/2 - \sqrt{2}/4$ (exercise: why?),

$$\Pr\left(\hat{p}_{\text{win}} < \frac{1}{1+\alpha} p_{\text{win}}\right) \leq 2e^{-\frac{\alpha^2 n}{C}}$$

for some large constant C.

So far we have managed to show that, except with probability exponentially small in n, provided the protocol does not abort in step 5. of the protocol it must be the case that $\hat{p}_{\text{win}} \geq p_{\text{win}}/(1+\alpha) \geq \cos^2 \pi/8 - 2\delta$ (if we choose $\alpha = \delta$). Now is time to apply the CHSH guessing lemma, Lemma 2.

The condition $p_{\text{id}} \leq 1/2 + 2(2\delta)^{1/2}$ that results gives a direct bound on the guessing probability of the device,

$$\mathrm{H}_{\min}(X_j|E) \geq -\log_2\left(\frac{1}{2} + 2(2\delta)^{1/2}\right) \geq 1 - C\sqrt{\delta} \tag{7.6}$$

for some small constant C.

Using the assumption that the device behaves identically and independently in each round of the protocol, the bound (7.6) does not only apply in the tested rounds $j \in T'$, but also in the rounds $j \in R$ used for the raw key. Thus as a final step we use (7.6) for $j \in R$, together with the fact that the devices are in tensor product form, to add up the entropies and conclude that $\mathrm{H}_{\min}(X_R|E) \geq |R|(1 - C\sqrt{\delta})$ — exactly what we set out to show!

A final subtlety is that this bound on the min-entropy holds under the conditions $|T'| \geq n/8$ and $\hat{p}_{\text{win}} \geq p_{\text{win}}(1-\delta)$. As we showed, conditioned on not aborting both conditions hold except with probability ε that is exponentially small. Taking this into account we obtain a lower bound on the smooth min-entropy of the raw key, $\mathrm{H}^{\varepsilon}_{\min}(X_R|E) \geq |R|(1 - C\sqrt{\delta})$. This bound is sufficient for privacy amplification to apply (the smoothing parameter ε will simply have to be added to the error of the extractor used for privacy amplification). Thus, provided that privacy amplification and information reconciliation are implemented correctly, Alice and Bob can generate a secure key.

7.3.2 Coherent attacks

The two-step approach we followed to analyze security against collective attacks no longer works against coherent attacks. First, since the devices may now have memory we cannot directly infer properties of the devices in the rounds used for the raw key from its behavior in the testing rounds. Second, since the global state prepared by Eve is no longer assumed to have a tensor product form we can no longer claim that the min-entropy adds up across rounds.

The first difficulty can be handled by using a variant of the concentration bound in Theorem 7.3.1 that applies to processes which may have memory, but still have a sequential nature and satisfy certain regularity properties. Such bounds are called martingale inequalities; one of the most useful is due to Azuma. By applying that inequality it is possible to obtain a similar bound as in (7.6) on the min-entropy per round for the raw key rounds from success of the CHSH test in the test rounds.

The second difficulty is more thorny. Given a lower bound $\mathrm{H}_{\min}(X_j|E) \geq h$ for some $h > 0$ for all $j \in R$, can we conclude a meaningful lower bound on $\mathrm{H}_{\min}(X_R|E)$? Unfortunately in general the answer is no: the quantum conditional min-entropy (in contrast to conditional von Neumann entropy) doesn't satisfy a nice form of the chain rule. To make progress we again need to use the sequential nature of our process. At this point there are different approaches to finishing the proof, and we mention just one, based on a technical result called "entropy accumulation theorem" (EAT) [DFR16]. The EAT gives conditions under which min-entropy "accumulates", and these conditions are satisfied by our setup. (The most important conditions are that the outputs are generated sequentially in each round, and are only a function of the state of the devices in that round; moreover the test, when it is performed, should be a deterministic function of the inputs and outputs in the round.)

Once it applies, the EAT is rather powerful, and it provides essentially the same consequences are we were able to derive in the case of collective attacks (except with a small loss in the parameter ε). Let's state the final result as a theorem.

Theorem 7.3.2 The DIQKD protocol, Protocol 1, satisfies the following properties. There is a $0 < \kappa \leq 1$ and $C \geq 1$ (depending on the tolerance parameter δ) such that the following hold for $\ell = \kappa n$ and $\varepsilon \leq 2^{-Cn}$.

First, there is an implementation of the devices such that the protocol does not abort with

probability at least $1-\varepsilon$.

Second, for any implementation of the devices, either the protocol aborts with probability larger than $1-\varepsilon$, or conditioned on not aborting Alice and Bob each produces a key of length ℓ such that $\Pr(K_A \neq K_B) \leq \varepsilon$ and

$$(1-\Pr(abort))D(\rho_{K_A E}, U_\ell \otimes \rho_E) \leq \varepsilon,$$

where E denotes all the side information available to the eavesdropper at the end of the protocol.

In our analysis we considered the min-entropy per round, and argued that it could be added up to obtain a bound on the min-entropy of the string x_R corresponding to Alice's raw key. A stronger bound can be obtained by using the fact that, when considering a large number of samples of a random variable X, the min-entropy converges to the von-Neumann entropy:

$$\frac{1}{n}\mathrm{H}^{\varepsilon}_{\min}(X_1 \cdots X_n) \approx_{n\to\infty} H(X)$$

for i.i.d. X, provided the smoothing parameter ε is chosen sufficiently large. This is called the "asymptotic equipartition property". Using this property it is possible to show that a lower bound on the von Neumann entropy in each round is enough to conclude a lower bound on the min-entropy of the whole string. Since the von Neumann entropy can in general be larger than the min-entropy this leads to better bounds on the key rate.

Acknowledgments

 The lecture notes are written by Nelly Ng, Thomas Vidick and Stephanie Wehner. We thank David Elkouss, Kenneth Goodenough, Jonas Helsen, Jérémy Ribeiro, and Jalex Stark for proofreading.

Bibliography

[Bha13] Rajendra Bhatia. *Matrix analysis*. Volume 169. Springer Science & Business Media, 2013 (cited on page 6).

[Che81] Herman Chernoff. "A note on an inequality involving the normal distribution". In: *The Annals of Probability* (1981), pages 533–535 (cited on page 12).

[DFR16] Frederic Dupuis, Omar Fawzi, and Renato Renner. "Entropy accumulation". In: *arXiv preprint arXiv:1607.01796* (2016) (cited on page 13).

[Eke91] Artur K Ekert. "Quantum cryptography based on Bell's theorem". In: *Physical review letters* 67.6 (1991), page 661 (cited on pages 3, 10).

[Pir+09] Stefano Pironio et al. "Device-independent quantum key distribution secure against collective attacks". In: *New Journal of Physics* 11.4 (2009), page 045021 (cited on page 9).

[Pir+10] Stefano Pironio et al. "Random numbers certified by Bell's theorem". In: *Nature* 464.7291 (2010), pages 1021–1024 (cited on page 11).

[VV14] Umesh Vazirani and Thomas Vidick. "Fully device-independent quantum key distribution". In: *Physical review letters* 113.14 (2014), page 140501 (cited on page 11).

Lecture Notes

Quantum Cryptography Week 8:

Quantum cryptography beyond key-distribution

Contents

In the last couple of weeks we discussed what are probably the most well-known quantum cryptographic protocols, namely the BB'84 and Ekert'91 protocols for quantum key distribution (QKD). However, quantum cryptography allows much more than just QKD. Quantum cryptographic protocols can be roughly divided into two categories. The first category concerns the use of quantum communication to implement classical tasks. An example of this is QKD, where the goal is to generate a classical key between two distant, but cooperating, parties. This week we will see other tasks which fall in this category. Contrary to QKD, however, this week the users implementing the task will not trust each other — so we will talk about "malicious Bob" playing against "honest Alice", and vice-versa.

The second category is concerned with genuinely quantum tasks, for which at least one of the inputs or outputs is quantum. An example of such a task is quantum secret sharing, where the goal is to distribute a qubit among several participants in such a way that at least a certain number of them need to come together to reconstruct the secret (we saw an example of a quantum secret sharing scheme in week 2).

This categorization does not cover all possibilities, and some important cryptographic tasks cannot be placed in either category. An example is delegated quantum computation. Here we imagine a user who has access to a powerful, but remote, quantum computer, while herself having only very limited, if any, quantum capabilities. The user's goal is to "delegate" a complex quantum computation she is interested in to the remote computer, without leaking any information about the computation or the input. We will learn more about delegated computation in week 10.

8.1 Two-Party cryptography

Let's imagine two parties, Alice and Bob, who can communicate over a classical, or even quantum, channel. Alice has some classical input x, which contains some private information only she has access to. Similarly Bob has a classical input y. Each of Alice and Bob would like to compute a certain function of both their inputs, $f_A(x,y)$ for Alice and $f_B(x,y)$ for Bob.

An easy solution would be for Alice and Bob to exchange their respective inputs and perform the computation locally. Unfortunately they don't trust each other: neither party wants to reveal more information about his or her input than is absolutely necessary. A typical example of such a task is "Yao's millionaire's problem". Here x and y represent Alice and Bob's respective fortune. The functions $f_A(x,y) = 1_{x>y}$ and $f_B(x,y) = 1_{y>x}$ will tell Alice and Bob if their fortune is larger than the others'. Can they decide who is the richest without announcing their actual fortune? (It turns out they can, but it's not so easy: we'll see how to do it later.)

Note that there is no eavesdropper here — this week we take a break from Eve. The communication channel between Alice and Bob is perfectly secure; the "adversary" is the other player rather than some external entity.

8.1.1 Secure Function Evaluation

The general task we just introduced is called Secure Function Evaluation (SFE). Let's formalize it.

Definition 8.1.1 Secure function evaluation (SFE) is a task involving two parties, Alice and Bob. Alice holds an input $x \in \mathscr{X}$ and Bob holds an input $y \in \mathscr{Y}$. Alice and Bob interact over a communication channel, and output an $a \in \mathscr{A}$ and $b \in \mathscr{B}$ respectively. We will say that a given protocol is a secure protocol computing a pair of functions $(f_A : \mathscr{X} \times \mathscr{Y} \to \mathscr{A}, f_B : \mathscr{X} \times \mathscr{Y} \to \mathscr{B})$ if it satisfies the following properties:

- *Correctness:* If both Alice and Bob follow the protocol (we say they are *honest*) then $a = f_A(x,y)$ and $b = f_B(x,y)$.
- *Security against cheating Bob:* If Alice is honest, then Bob cannot learn more about her

input x than he can infer from $f_B(x,y)$.
- *Security against cheating Alice:* If Bob is honest, then Alice cannot learn more about his input y than she can infer from $f_A(x,y)$.

Note that the definition does not guarantee anything in case Alice and Bob are both dishonest. In this case there is nothing we can do! The goal is only to protect the honest parties.

The definition we gave is rather informal — what does it mean that "Bob cannot learn more about Alice's input x than he can infer from $f_B(x,y)$"? It turns out that this requirement is rather tricky to make precise. We will return to it in a moment. First let's consider some examples that illustrate the kind of problems that fit in this framework.

■ **Example 8.1.1** Alice and Bob are contemplating going to a movie. Here, $x,y \in \{0,1\}$ where '0' denotes "no" and '1' denotes "yes". The function they wish to compute is

$$f(x,y) = f_A(x,y) = f_B(x,y) = x \text{ AND } y.$$

Let us see what security means here. If $f(x,y) = 1$, then it must be that $x = y = 1$ and both parties learn the other's input. Alice and Bob go to the movies. If $f(x,y) = 0$, then it must be that either $x = 0$ or $y = 0$ (or both). If Alice input $x = 1$, i.e., she would like to go to a movie, and the output is $f(x,y) = 0$, then Alice can infer $y = 0$, but Bob will never learn whether $x = 0$ or $x = 1$. So a party only learns the others' input if they themselves declared they wanted to go to a movie. ■

■ **Example 8.1.2** Alice (a customer) wants to identify herself to Bob (an ATM). Here, x is the password honest Alice should know, and y the password (for Alice) that the honest ATM should have stored in its database. The function f is the equality, that is, $f(x,y) = 1$ if and only if $x = y$, and $f(x,y) = 0$ otherwise. Security means that if Alice is dishonest (she might not know x but is still trying to break through the ATM's authentication system), then Bob should have the guarantee that Alice will never learn anything more about his input y than she can infer from $f(x,y)$ — that is, whatever x she tries, that $x \neq y$! (Unless she happens to be lucky of course.) Similarly, if Bob is a fraudulent ATM who is out to steal passwords from the users, the best he can do is guess a y and see whether it worked. No more information is revealed. ■

■ **Example 8.1.3** Alice wants to sell a book to Bob. Here, x is Alice's asking price, and y is Bob's bid. The function they wish to compute is $f(x,y) = (ok,y)$ if $y \geq x$, and $f(x,y) = (no,0)$ if $y < x$. If $f(x,y) = (ok,y)$, Alice can proceed to sell the book to Bob. Bob pays what he offers, and Alice gets at least her asking price. Security means that dishonest Bob can never learn what the asking price actually was, only that it was less or equal than his bid. If $f(x,y) = (no,0)$, then Alice will not sell her book. Security means that Alice will never learn exactly what Bob's bid actually was, only that it was lower than her asking price. Similarly, Bob will only learn that Alice's asking price was higher than his bid. ■

8.1.2 The simulation paradigm for security

We won't attempt to give a fully formal definition of security of SFE, as this would take us too far. Good references to learn more on the topic include a survey by Goldreich [Gol05] (especially Section 7) and one by Lindell [LP09]. For the case of quantum protocols, it is only recently that a satisfactory definition has been introduced; see [Unr10], or [FS09] for a weaker but perhaps more approachable definition.

The key concepts based on which security is defined are those of *ideal functionality* and *simulators*.

Definition 8.1.2 — Ideal functionality. Let $(f_A : \mathscr{X} \times \mathscr{Y} \to \mathscr{A}, f_B : \mathscr{X} \times \mathscr{Y} \to \mathscr{B})$ be a pair of functions corresponding to an SFE task. The *ideal functionality* is a device which takes as input $x \in \mathscr{X}$ and $y \in \mathscr{Y}$, and returns $f_A(x,y)$ and $f_B(x,y)$.

Thus the ideal functionality does precisely what a protocol solving the SFE task is supposed to achieve: it directly returns the values of each of the two functions. It is "ideal" in the sense that it does not require any interaction between the two parties: you should picture a "black box" which takes the inputs and provides the outputs, no questions asked. Informally, we will then say that a protocol for SFE is secure if, provided one of the parties is honest, whatever the other party does there is nothing more they can produce that they could not have produced by interacting with the ideal functionality.

■ **Example 8.1.4** Consider again the millionaire's problem. Here the ideal functionality takes as input x from Alice and y from Bob, and returns $f_A(x,y) = 1_{x>y}$ to Alice and $f_B(x,y) = 1_{y>x}$ to Bob. Now, suppose we are given a protocol for this problem, and suppose a malicious Bob was able to infer Alice's fortune x through his interaction with her. Then the simulation paradigm dictates that, if the protocol was secure, he should be able to do the same through an interaction with the ideal functionality. But the ideal functionality just takes any y' of Bob's choice and returns to him $f_B(x,y') = 1_{y'>x}$. Since only one interaction is allowed, the best Bob can do is find out if $x < y'$ for a single y' of his choice, something which for this particular SFE task is unavoidable. ■

A little more formally, we make the following definition.

Definition 8.1.3 — Security against cheating Bob. A protocol for an SFE task (f_A, f_B) is secure if for any malicious Bob interacting with an honest Alice in the protocol, there exists a *simulator* which, by controlling Bob in an interaction with the ideal functionality (where Alice acts as honest Alice in the protocol), is able to generate a distribution on outputs that is indistinguishable from the distribution produced by malicious Bob in the real interaction.

(A symmetric definition can be given for security against cheating Alice.) The definition refers to the output distributions being "indistinguishable" from one another. This by itself is a subtle notion. Indistinguishability can be either "statistical", meaning that the output distributions are close in total variation distance (the classical analogue of the trace distance), or "computational", meaning that the distributions cannot be distinguished by any efficient (polynomial-time) algorithm. In the classical setting most interesting tasks in two-party cryptography cannot be proven secure for the stronger notion of statistical indistinguishability, so that one has to rely on computational indistinguishability.

In these notes our main focus is on finding certain tasks for which statistical indistinguishability can be achieved by quantum protocols, but not by classical ones. We won't worry too much about giving formal proofs of security, but you should be aware that doing so can be quite tricky. In fact, many *wrong* proofs of security of quantum protocols have been given in the past by relying on "intuitive" arguments rather than the simulation paradigm. We will point out one such example later.

8.2 Oblivious Transfer

Let's discuss an important example of an SFE task, *Oblivious Transfer* (OT). Here $\mathscr{X} = \{0,1\}^\ell \times \{0,1\}^\ell$ and $\mathscr{Y} = \{0,1\}$. That is, Alice receives as input two ℓ-bit strings $x = (s_0, s_1)$, and Bob receives as input a single bit y. The goal is for Bob to obtain $f_B(x,y) = s_y$, while Alice will obtain nothing, $f_A(x,y) = \perp$. Thus a protocol will implement this task securely if, first of all it is correct, and second, malicious Alice will not obtain any information at all about Bob's input bit (since the ideal functionality never returns anything to Alice), and malicious Bob will at best learn one of Alice's strings, but never more.

As an example scenario where this task could be useful, imagine that Alice is a database which contains two entries s_0 and s_1. Bob would like to retrieve one of them, but does not want Alice to know which one he retrieved, preserving his privacy. Alice can also be sure that he does not retrieve her entire database.

What makes OT truly interesting is the fact that it is *universal* for two-party cryptography. That is, if we can build a secure protocol for 1-2 OT then we can solve *any* SFE problem by just using 1-2 OT multiple times. In this respect 1-2 OT is analogous to a universal gate in computing. This universality property can be shown in different ways; see Section 6 in [PS10] for a construction based on secret sharing called "Yao's garbled circuits".

Unfortunately it is also known that OT cannot be implemented securely without computational assumptions... in the classical world. What about quantum protocols? Let's consider the following natural protocol, introduced in [Ben+91].

Protocol 1 — Quantum OT protocol. Alice has input $x = (s_0, s_1) \in \{0,1\}^\ell \times \{0,1\}^\ell$ and Bob has input $y \in \{0,1\}$. Their goal is to compute $(f_A(x,y) = \perp, f_B(x,y) = s_y)$.

1. Alice selects uniformly random $x \in \{0,1\}^{2n}$ and $\theta \in \{0,1\}^{2n}$. She prepares BB'84 states $|x_j\rangle_{\theta_j}$ for $j = 1,\ldots,2n$ and sends them to Bob.
2. Bob measures each of the qubits he received from Alice in a random basis $\tilde{\theta}_j$, obtaining outcomes $\tilde{x}_1,\ldots,\tilde{x}_n \in \{0,1\}$. He notifies Alice that he is done with his measurements.
3. Alice reveals her choice of bases $\theta_1,\ldots,\theta_{2n}$ to Bob.
4. Bob sets $I = \{i : \theta_i = \tilde{\theta}_i\}$, $I_y = I$ and $I_{1-y} = \{1,\ldots,2n\}\backslash I$. (For simplicity, assume that $|I_0| = |I_1| = n$.) Bob sends (I_0, I_1) to Alice.
5. Alice sends $t_0 = s_0 \oplus x_{I_0}$ and $t_1 = s_1 \oplus x_{I_1}$ to Bob.
6. Alice outputs $\perp$, and Bob outputs $t_y \oplus \tilde{x}_{I_y}$.

Let's first check that this protocol is correct. This is clear: whenever $j \in I_y$, by definition $\theta_j = \tilde{\theta}_j$, therefore $x_j = \tilde{x}_j$ and $(s_y \oplus x_{I_y}) \oplus \tilde{x}_{I_y} = s_y$.

Is it secure? Security against cheating Alice is not hard to verify. Indeed, the only information she gets from a honest Bob are two sets (I_0, I_1). If Bob is honest, even if Alice sent him arbitratry states in the first step, and misleading basis information in the third, since Bob's choice of $\tilde{\theta}_j$ is uniformly random the sets I_0, I_1 will be a uniformly random partition of $\{1,\ldots,2n\}$ that contains no information at all about his input y. So anything a dishonest Alice could do in this protocol can be simulated by an interaction with the ideal functionality, where the simulator would replace Bob's message (I_0, I_1) (which is not provided by the ideal functionality) with a uniformly random choice.

How about security against cheating Bob? The idea is supposed to be that, given Bob's basis choices are random, he can at best learn roughly half of Alice's inputs $\tilde{x}_j$. Of course he could lie about which half he learned, but in any case he will only be able to recover about half of the bits of Alice's input $x = (s_0, s_1)$. Note he could still, for example, learn half of s_0 and half of s_1 (instead of the whole s_0 or s_1 and nothing about the other). This can be prevented by adding in a layer of privacy amplification (recall from Week 4) to the protocol, so let's assume it is not a serious issue.

You might already have noticed there is a more worrisome hitch. The protocol requires Bob to "measure each qubit he received from Alice", and then "notify Alice that he is done with his measurements". But what if Bob is malicious — what if he stores Alice's qubits in a large quantum memory, without performing any immediate measurement, and lies to her by declaring that he is done? Alice would then naively reveal her basis information, and Bob could measure all the qubits he stored using $\tilde{\theta}_j = \theta_j$. He would thus obtain outcomes $\tilde{x}_j = x_j$ for all j, and he could recover both $s_0 = t_0 \oplus \tilde{x}_{I_0}$ and $s_1 = t_1 \oplus \tilde{x}_{I_1}$!

So the protocol we gave is not at all secure. There are two ways to get around the problem. One possibility is to make certain physical assumptions on the capacities of cheating Bob. A possible assumption is that Bob has a bounded quantum memory, in which case he wouldn't be able to store all of Alice's qubits. We will explore this assumption next week. Another possibility would be to somehow force Bob to *commit* to a choice of basis $\tilde{\theta}_j$, and outcomes $\tilde{x}_j$ that he obtained, *before* Alice would accept to reveal her θ_j. Of course, to avoid reversing the difficulty it should be that Alice cannot learn any information about the $\tilde{\theta}_j$ just from Bob's commitments. The task we're

trying to solve is called *bit commitment*, and it is another fundamental primitive of multi-party cryptography. Let's explore it next.

8.3 Bit commitment

The goal of a bit commitment protocol is to provide a means for Alice to *commit* to an unbreakable promise, without revealing any information about the promise to Bob until Alice decides to *open* her promise — without being allowed to change her mind in-between the "commit" and "open" phases.

Definition 8.3.1 — Bit Commitment (BC). Bit commitment is a task involving two parties, Alice (the committer) and Bob (the receiver). The input to Alice is a single bit $b \in \{0,1\}$, and she has no output. Bob has no input, and his output is b'. A protocol for bit commitment has two phases, the *commit phase* and the *open phase*, and it should satisfy the following properties:

1. (Correctness) If both Alice and Bob are honest then at the end of the protocol Bob outputs a bit $b' = b$.
2. (Hiding) For any malicious Bob, the state of Bob at the end of the commit phase (including all his prior information and information received from Alice during the commit phase, classical or quantum) is independent of b.
3. (Binding) For any three possible malicious behavior A, A_0 and A_1, the probabilities p_b that Bob outputs $b' = b$ after interacting with A in the commit phase and A_b in the open phase satisfy $p_0 + p_1 \leq 1$.

The hiding property is clear: it states that, after the open phase, Bob still has no information at all about the bit b that honest Alice committed to. The binding property is more subtle. Intuitively, what it is trying to capture is that once Alice has committed to a specific value b (this is the role of A in the definition), then she shouldn't be able to come up with two possible different behavior (A_0 and A_1) such that she has a strictly higher than $1/2$ chance of being able to convince Bob that $b = 0$ (she would run A_0) *or* that $b = 1$ (she would run A_1).

Exercise 8.3.1 Give a secure protocol for Yao's millionnaire's problem, assuming you have access to a protocol securely implementing bit commitment. ∎

Bit commitment is a good example of a cryptographic task for which it is crucial to define security as precisely as possible, especially in the quantum setting. Consider the following "intuitive" definition of the binding property: "It should be impossible for malicious Alice to convince honest Bob that $b = 0$ *and* $b = 1$ with probability strictly larger than 1". Do you see the difference? I wouldn't blame you if you didn't — the pioneers of quantum information and cryptography didn't either! In 1991 Brassard et al. [Bra+93] famously proposed an "unconditionally secure" quantum protocol for bit commitment, that satisfied the above intuitive notion of security. However, their protocol was later completely broken! (Indeed, as we will soon see, perfectly secure bit commitment is impossible both in the classical and the quantum world.) Their "mistake" is that they interpreted the italicized "and" in the intuitive definition above in a strong sense: they show that, in their protocol, it wouldn't be possible for a malicious Alice to simultaneously convince Bob that $b = 0$ and $b = 1$, by assuming that, if this where the case, the two final quantum states of the protocol associated with the outcomes "Bob returns $b' = 0$" and "Bob returns $b' = 1$" would exist simultaneously. However, as we know very well by now, quantum information is subtle, and the fact that Alice can "change her mind" after the commit phase does *not* imply that she can generate both the $b' = 0$ and $b' = 1$ states for Bob from the same state at the end of the open phase; only that she can generate either of them.

8.3.1 Universality of bit commitment

Bit commitment is an important task in quantum multiparty cryptography because, just as OT, it is known to be universal. This is demonstrated by the protocol for OT we gave in the previous section: as we discussed, the protocol by itself is not secure; however if one has access to a secure protocol for bit commitment then it can be turned into a secure protocol as well. Since OT itself is universal for multiparty computation we deduce that bit commitment is universal. However, note that the protocol for OT based on bit commitment we gave is quantum, even if bit commitment is implemented using a classical protocol. It is interesting to note that this is unavoidable: indeed, bit commitment is *not* universal for *classical* multiparty computation!

Let's see how the reverse can be accomplished, using OT as a building block to achieve bit commitment. For this, we will consider an approximate version of bit commitment, in which Alice can change her mind with some small error probability ε. That is, the protocol is ε-binding in that the requirement $p_0 + p_1 \leq 1$ is relaxed to $p_0 + p_1 \leq 1 + \varepsilon$.

The following protocol takes 1-2 OT and turns it into bit commitment. We will invert the use of 1-2 OT: Bob will now be the sender, and Alice the receiver.

Protocol 2 — Bit commitment from 1-2 OT. Alice's input is $b \in \{0,1\}$. Bob has no input.

1. Commit phase: Bob chooses two strings $s_0, s_1 \in \{0,1\}^\ell$ uniformly at random. Bob and Alice execute a protocol for OT, with the role of the players reversed: OT-Alice's input is (s_0, s_1) (provided by Bob), and OT-Bob's input is b (provided by Alice). Thus Alice receives s_b, and Bob receives $\perp$.
2. Open phase: Alice sends $\hat{b}$ and $\hat{s} = s_b$ to Bob. If $\hat{s} = s_{\hat{b}}$, then Bob accepts and concludes Alice committed herself to $b = \hat{b}$. If $\hat{s} \neq s_{\hat{b}}$, then Bob rejects.

Why does this give bit commitment? First of all, if both parties behave honestly the protocol is clearly correct. Let's consider the hiding property. We need to show that, at the end of the commit phase, Bob has no information about b. This follows right away from the definition of OT, which guarantees that the sender never receives any information about the receiver's input.

It remains to show the protocol is ε-binding. This again follows from the security of OT, for $\varepsilon = 2^{-\ell}$. Indeed, the ideal functionality for OT is such that the receiver can learn only *one* of the two strings. Suppose Alice has two possible strategies, one to open $\hat{b} = 0$ and the other to open $\hat{b} = 1$. Let p_0 be the probability that the first strategy succeeds, and p_1 the probability that the second succeeds. As a consequence, Alice can recover both of s_0 and s_1 with probability at least $p_0 + p_1 - 1$. By the security of the OT primitive, this can happen with probability at most the probability that a random guess of the non-received string would succeed, i.e. $2^{-\ell}$. By taking ℓ large enough we can achieve any desired ε-security for the binding property.

If you have been reading carefully you may have noted that in the argument above we made a jump from "Alice can recover s_0 with probability p_0, and s_1 with probability p_1" to "Alice can recover both of s_0 and s_1 with probability at least $p_0 + p_1 - 1$". While this is correct if Alice is classical, if her strategies involved incompatible quantum measurements the implication might no longer be true. Hence in case we allow the protocol implementing OT to be a quantum protocol, we have to be additionally careful in showing that the resulting protocol for bit commitment satisfies the required definition. This is possible (so the protocol described above *is* secure provided the implementation of OT is, whether classical or quantum) but one must take even greater care in making the right security definitions to ensure that they satisfy the stringent criteria of "universal composability" [Unr10].

8.3.2 Impossibility of bit commitment

Since, as we argued, bit commitment implies OT (in the quantum world), but perfect OT is impossible, it must be that bit commitment is impossible as well! Let's see why. We give an

informal argument, and refer you to the detailed notes by Watrous on the subject [Wat06] for a more rigorous proof.

Consider a protocol for bit commitment that is perfectly hiding, i.e. at the end of the commit phase Bob has absolutely no information about Alice's bit b. Let's show that in this case a malicious Alice can cheat *arbitrarily*: she is able to open any bit that she likes.

Suppose that the initial state of Alice and Bob is a pure state $|\psi\rangle_{AB}$, which in particular contains Alice's input. We can always assume this is the case by considering a purification and giving the purifying system to Alice.

Now suppose Alice executes the bit commitment protocol with input b. At the end of the commit phase, the joint state of Alice and Bob can be described by some pure state $|\psi(b)\rangle_{AB}$. Since the bit commitment protocol is perfectly hiding, it must be the case that

$$\rho_B(0) = \mathrm{tr}_A(|\Psi(0)\rangle\langle\Psi(0)|_{AB}) \quad \text{and} \quad \rho_B(1) = \mathrm{tr}_A(|\Psi(1)\rangle\langle\Psi(1)|_{AB})$$

are absolutely identical, as otherwise there would be a measurement that Bob can make on his system to distinguish (even partially) between the two states, giving him some information about b.

Now is time to take out our quantum information theorist's toolbox and extract one of its magic tools: Uhlman's theorem! The theorem implies that, if $\rho_B(0) = \rho_B(1)$, then necessarily there exists a unitary U_A on Alice's system such that $U_A \otimes \mathbb{I}_B|\Psi(0)\rangle_{AB} = |\Psi(1)\rangle_{AB}$. But this means Alice can perfectly change her mind, thereby completely breaking the binding property for the protocol.

Rather unfortunately for the fate of quantum multiparty cryptography, it is possible to generalize this argument to show that any protocol for *any* task in multiparty quantum cryptography must be "totally insecure" in the following sense: if the protocol is perfectly secure against a malicious Bob, then it must be that a malicious Alice (interacting with honest Bob) can recover the value $f_A(x,y)$ associated with Bob's input y for *all* possible values of x, simultaneously! (To see why this indeed renders the protocol totally insecure, consider for example the millionnaire's problem: Alice would learn if $x > y$ for any x, and could thus perform a quick binary search to learn Bob's fortune y exactly.)

Given such a strong impossibility result, due to [LC97; May97], we are left with two possibilities. Either we place limiting assumptions on any malicious player's abilities. In classical cryptography these are mostly computational assumptions, and we give an example in the next section. In quantum cryptography a very successful approach considers physical assumptions on the adversary, such as it having limited storage capabilities. We will explore such assumptions in detail next week.

The second option consists in taking act of the impossibility of *perfect* protocols for multiparty cryptography and instead settle for protocols with a relaxed notion of security, where e.g. Bob can learn "some" information about both Alice's input strings s_0, s_1 in bit commitment, but not all. This is indeed possible, and can be quite useful in spite of the relaxed security condition. We'll see an example in Section 8.4.

8.3.3 Computationally secure commitments

We have seen that it is impossible to perfectly implement bit commitment, whether we use quantum information or not. The fact that it is such a useful primitive, however, should encourage us to be creative. In many contexts we would be willing to put up with our usual requirement for perfect, information-theoretic security, and start making assumptions — of course, the fewer the better! One possibility is to make physical assumptions, such as that the malicious party has a bounded amount of quantum memory. We will discuss this assumption in much more detail next week.

We can also assume that the malicious party has bounded computational power. This is a very standard assumption in classical cryptography, as indeed very little can be achieved without it (in

contrast to quantum cryptography). Of course, here we would only want to make assumptions that hold even if the malicious party has bounded *quantum* computational power. The weakest such assumption under which any interesting cryptographic task is made possible is the existence of *one-way functions*. Informally, a function is one-way if it is easy to evaluate the function on any input (there is an efficient algorithm to compute it), but it is hard to invert the function (given a point in the range of the function, find a pre-image).

There are many candidate constructions of one-way functions, including some that are believed to be hard to invert even for quantum computers. And it turns out that, assuming one-way functions exist, there is a simple protocol for bit commitment that is statistically binding ($p_0 + p_1$ can be made as close to 1 as desired by increasing the amount of communication required in the scheme), and computationally hiding (the hiding property holds as long as it can be assumed that the malicious party cannot invert the one-way function). For a description of the protocol we refer you to the videos; you may also be interested in Section 4.7 in the lecture notes [PS10], which presents a different protocol based on the (somewhat stronger) assumption of existence of one-way permutations, and discusses applications of bit-commitment to zero-knowledge proof systems.

8.4 Coin flipping

Let's see a last example of an interesting two-party task: coin-flipping. This problem was introduced by Blum [Blu83]: imagine Alice and Bob want to flip a fair coin to determine who has to do some chore — for example, prepare the next lecture. But Alice is in Europe, Bob is in North America, so they have to do this over the phone. Is there a good protocol, that would ensure both Alice and Bob obtain the same outcome for the coin flip, but such that neither can bias it one way or the other?

Definition 8.4.1 — Strong coin flipping. In the task of *coin flipping* there are two players, Alice and Bob. Neither has an input. The goal is for both players to output the same value $c \in \{0,1\}$ such that the following properties hold.

- Correctness: if both Alice and Bob are honest then c is uniformly distributed.
- ε-secure: neither player can force $p(c=0) \geq 1/2+\varepsilon$ or $p(c=1) \geq 1/2+\varepsilon$, where $p(c)$ is the probability that the honest player outputs a value c.

The smallest ε for which a protocol is ε-secure is called the *bias*.

Coin flipping does not fall in the framework of SFE, because it is a randomized primitive: there is no fixed function of the players' inputs that determines their outputs (indeed, here they do not have any input at all). Thus the strong impossibility results that we saw for SFE, both in the classical and quantum setting, no longer apply...is perfectly secure coin flipping possible?

8.4.1 Classical coin flipping

Let's see if the following simple protocol does the trick.

Protocol 3 — Blum coin flipping.

1. Alice flips a random bit $a \in \{0,1\}$ and sends it to Bob.
2. Bob flips a random bit $b \in \{0,1\}$ and sends it to Alice.
3. Both players return $c = a \oplus b$.

Is this protocol secure? It is certainly correct: if both players are honest then c is uniformly distributed. In fact, it is sufficient that one player is honest: as long as a or b is random then $a \oplus b$ will be random. Or will it? Note that the protocol forces us to specify an order in which the players exchange their messages (indeed, it is never wise to attempt to speak simultaneously over the phone). Here we made Alice go first, and Bob second. So Bob receives Alice's message a before he sends her his choice of b. But then he can easily force any outcome b' of his choice by setting $b = b' \oplus a$: the protocol is completely insecure.

Unfortunately this is the fate of *any* classical protocol for coin flipping: no value of $\varepsilon < 1/2$ can be achieved for security! That is, if one player cannot completely bias the outcome of the protocol to a certain value, then the other player can: there is always at least one of Alice or Bob who can perfectly cheat. Informally, the reason is the same that makes the Blum protocol insecure: one can argue that, whatever the outcome c of the protocol, it has to be determined at *some* point in the protocol. By considering the messages exchanged from the last to the first, one can find a message such that, before the message is sent the outcome is not yet determined, but once the message has been sent it is (in the case of the Blum protocol, this would be Bob's message). But then whomever sends that message has the ability to bias the outcome to any possibility.

8.4.2 Quantum coin flipping

The impossibility argument given in the previous section does not immediately apply to quantum protocols, as for a quantum protocol the notions of a transcript, and the outcome being determined, are much less clear: everything can happen in superposition. And for once, there is good reason: strong coin-flipping is possible using quantum information, at least for some values of $\varepsilon < 1/2$. Let's see an example protocol, discovered by Aharonov et al. [Aha+00].

Protocol 4 — ATVY coin-flipping. For $a,x \in \{0,1\}$ define the qutrit

$$|\phi_{a,x}\rangle = \frac{1}{\sqrt{2}}|0\rangle + (-1)^x|a+1\rangle.$$

1. Alice selects $x \in \{0,1\}$ and $a \in \{0,1\}$ uniformly at random and sends $|\phi_{a,x}\rangle$ to Bob.
2. Bob selects $b \in \{0,1\}$ uniformly at random and sends b to Alice.
3. Alice sends a and x to Bob.
4. Bob verifies the state he received from Alice in step 1. is $|\phi_{a,x}\rangle$ (e.g. by measuring in any orthonormal basis containing $|\phi_{a,x}\rangle$). If it is not the case then he declares that Alice has been cheating and aborts the protocol.
5. Both players return the outcome $c = a \oplus b$.

Note the similarity between this protocol and the Blum protocol we saw earlier. Here as well Alice and Bob each choose "half" of the outcome c: Alice chooses a, Bob b, and they return $c = a \oplus b$. However, Alice does not fully reveal a to Bob in her first message: instead, she provides him with some form of "weak commitment" to a in the form of the state $|\phi_{a,x}\rangle$. Because the four states $|\phi_{a,x}\rangle$ are not orthogonal, it is impossible for Bob to completely discover the value of a without being revealed x first, which only happens after he has had to make his choice of b.

Exercise 8.4.1 Compute the reduced density matrices $\rho^B_{|a=0}$ and $\rho^B_{|a=1}$ associated with Bob's view of the protocol after Alice's first message has been sent, for a uniformly random choice of $x \in \{0,1\}$. Show that the the trace distance between these two matrices is $1/2$, and conclude that the probability with which Bob can force an outcome c of his choice is at most $3/4$. ∎

The exercise shows that the maximum bias that a cheating Bob can induce in the protocol is $\varepsilon = 1/4$. Security for cheating Alice is a bit harder to argue, because we have to consider the possibility for her to prepare an arbitrary state in the first step, which may be entangled with some information she keeps on the side and uses, together with the value b received from Bob, to determine her message in the third step of the protocol. We will not give the details here, but the result is the following:

Theorem 8.4.1 — (Amb01). The ATVY coin-flipping protocol is correct and ε-secure for $\varepsilon = 1/4$.

Can we do even better? Unfortunately it turns out that perfectly secure strong coin-flipping is also impossible for quantum protocols: Kitaev showed that the smallest bias any protocol could achieve is $\varepsilon = (\sqrt{2}-1)/2 \approx 0.207$. Kitaev's proof is an extension of the classical impossibility argument, based on an ingenuous representation of transcripts for quantum protocols; see [Amb+04] for details. If you are interested in Section 8.5 below we give a "dual" argument to Kitaev's, based on [GW07].

The good news, though, is that Kitaev's bound is achievable: for any $\varepsilon > 0$ there is a quantum strong coin-flipping protocol with bias $(\sqrt{2}-1)/2+\varepsilon$. The protocol achieving this, however, is rather complex. It is based on the notion of *weak coin flipping*, that we take a look at next.

8.4.3 Weak coin flipping

The task of weak coin flipping is defined as strong coin flipping, except the security requirement is weaker: instead of requiring that neither player can force $p(c=0) \geq 1/2+\varepsilon$ or $p(c=1) \geq 1/2+\varepsilon$, we only require that malicious Alice cannot force $p(c=0) \geq 1/2+\varepsilon$, and malicious Bob cannot force $p(c=1) \geq 1/2+\varepsilon$. That is, we assume a priori that the malicious behavior of each player will always try to achieve a certain pre-determined outcome. For instance, in the example we used earlier, the outcome of the coin flip determines who has to accomplish a certain chore: if $c=0$ it will be Bob, and if $c=1$ it will be Alice. In this scenario the only thing we're really worried about is that Alice would manage to increase the probability that $c=0$, while Bob would increase the probability that $c=1$. So all we need is a weak coin flipping protocol.

It turns out that weak coin flipping, with arbitrarily small (but non-zero) bias ε, is indeed possible [Moc07]. The best protocol known for achieving this, however, remains very complex, and requires a large number of rounds of interaction, that scales exponentially with $1/\varepsilon$ [Aha+16]. (It is known that some dependence on $1/\varepsilon$ is necessary, but it is an open problem to do better than exponential.) Chailloux and Kerenidis [CK09] showed that any weak coin flipping protocol with bias ε could be used to build a strong coin flipping protocol with bias $(\sqrt{2}-1)/2+O(\varepsilon)$, thereby matching Kitaev's lower bound.

8.5 Kitaev's lower bound on strong coin flipping

In this section we give a proof of Kitaev's lower bound on the bias of secure protocols for strong coin flipping, based on [GW07]. The argument relies on simple notions of linear and semidefinite programming with which you may not already be familiar. If so we encourage you to read a bit about these techniques, as the proof is very elegant and well worth understanding. But if you don't have the time then you can skip the section: it is not required for the course.

For any coin-flipping protocol, define p_{1*} as the probability that Alice outputs a 1, maximized over all possible (cheating) strategies for Bob. Define p_{*1} symmetrically. Then the condition for the protocol to be a secure strong coin-flipping protocol with bias ε is that both $p_{1*}, p_{*1} \in [1/2-\varepsilon, 1/2+\varepsilon]$. (This is equivalent to the condition $p_{1*}, p_{*0} \in [1/2-\varepsilon, 1/2+\varepsilon]$ we introduced earlier.)

Theorem 8.5.1 — Kitaev. For any strong coin-flipping protocol, we have $p_{1*}p_{*1} \geq \frac{1}{2}$.

Note that the condition in the theorem immediately implies that any strong coin flipping protocol has bias at least $(\sqrt{2}-1)/2$, as claimed.

Kitaev's theorem applies to both classical and quantum protocols. We'll see the proof for classical protocols first, and then move to the quantum setting. Both proofs have the same structure: Bob's maximum cheating probability can be expressed as the optimum of a linear program (LP) (or semidefinite program (SDP) in the quantum case), and similarly for Alice's. Any feasible solution to the duals of each LP provides an upper bound on the probability of success of the cheating strategy.

The crucial insight is that the cheating probabilities need to be considered *together*, through the quantity $p_{1*}p_{*1}$: a good upper bound on this quantity expresses the fact that, either Alice can force Bob to output a 1, or, if she can't, then it must be that Bob can force her to produce a 1. We will obtain a bound on this bias by taking the product of some of the dual LP (or SDP) constraints. Let's proceed with the details.

8.5.1 The bound on classical protocols

Fix a classical protocol. We can think of the protocol as a tree, where each node is indexed by a variable u representing the transcript that led to this node: if we are in node u, and Alice plays by sending a message a, then we arrive at node (u,a). The honest protocol is given by probabilities $p_A(a|u)$, $p_B(b|u)$, which are Alice's (resp. Bob's) transition probabilities. Given that Alice is honest, Bob's maximum cheating probability can be expressed as a linear program LP_B, in which the variables $p_B(u)$ represent the probability of reaching node u, when Alice is honest and Bob cheats. Bob's goal is to maximize the probability of reaching a leaf labeled with a 1; denote this set L_1. We introduce constraints to express the fact that Bob can choose any distribution on edges when it is his turn to play, but he has to follow Alice's distribution when it is her turn.

$$
\begin{aligned}
(\mathrm{LP}_B,\ \text{primal}) \qquad \max \quad & \sum_{u\in L_1} p_B(u) & \\
& p_B(u)p(a|u) = p_B(u,a) && \forall a, \forall u \text{ node for Alice} \\
& p_B(u) = \sum_b p_B(u,b) && \forall u \text{ node for Bob} \\
& p_B(0) = 1 & \\
& p_B(u) \geq 0 && \forall u
\end{aligned}
$$

To write the dual of this linear program, introduce variables $Z_A(u,a)$ for the first set of constraints and $Z_A(u)$ for the second set. With a little work the dual can be written in the form

$$
\begin{aligned}
(\mathrm{LP}_B,\ \text{dual}) \qquad \min \quad & Z_A(0) & \\
& Z_A(u) \geq \sum_a p(a|u) Z_A(u,a) && \forall u \text{ node for Alice} \\
& Z_A(u) \geq Z_A(u,b) && \forall b, \forall u \text{ node for Bob} \\
& Z_A(u) \geq 1 && \forall u \in L_1
\end{aligned}
$$

$Z_A(u)$ can be interpreted as the maximum probability with which Bob can cheat, starting at node u. We can consider another linear program LP_A, this time for a cheating Alice, which is completely symmetrical. The interpretation of the variables Z_A, Z_B motivates the introduction of the quantity

$$F_\ell = \mathrm{E}_{u\sim\ell}\big[Z_A(u)Z_B(u)\big] \tag{8.1}$$

where $u \sim \ell$ is shorthand for u being taken according to the probability distribution on nodes at depth ℓ which arises from the honest protocol. In this expression, $Z_A(u)Z_B(u)$ should be interpreted as the bias that cheating players can achieve, if any of them starts cheating at node u.

Let Z_A, Z_B be optimal solutions to the duals of LP_B and LP_A respectively. The last constraint of the dual implies that without loss of generality we can assume that both Z_A and Z_B are both exactly 1 at all leaves labeled with a 1 (as if they were larger, a better solution to the LP could be obtained by scaling). Hence if n is the last level of the game, then $F_n = p_{1,1} = 1/2$. Moreover, strong duality implies that $F_0 = p_{1*}p_{*1}$. Finally, by multiplying out the constraints of the two duals one easily gets that $F_\ell \geq F_{\ell+1}$, which proves Theorem 8.5.1 for the case of classical protocols.

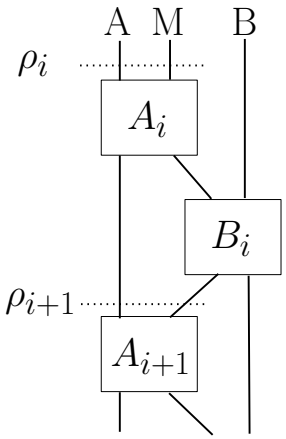

Figure 8.1: Step i in the coin flipping protocol

8.5.2 The bound on quantum protocols

For quantum protocols the bound follows analogously, with a few tweaks. A protocol is modeled by a series of unitary operations A_i for Alice and B_i for Bob. The players are assumed to start in the $|0\ldots0\rangle$ state. At the end of the interaction, they measure using $\{\pi_A, \mathbb{I}-\pi_A\}$, $\{\pi_B, \mathbb{I}-\pi_B\}$ respectively and return the outcome. The correctness requirement is that

$$p_{1,1} = \|(\pi_A \otimes \mathbb{I}_M \otimes \pi_B) B_n A_n \cdots B_1 A_1 |0\ldots0\rangle\|^2 = 1/2$$

and $p_{0,0}$, defined symmetrically using $(\mathbb{I}-\pi_A), (\mathbb{I}-\pi_B)$ as the measurements, is also $1/2$. The following SDP captures the maximum cheating probability for Bob:

$$\begin{aligned}
(\text{SDP}_B,\ \text{primal}) \qquad & \max \quad \langle \pi_B \otimes \mathbb{I}, \rho_n \rangle & \\
& \text{Tr}_M(\rho_{i+1}) = \text{Tr}_M(A_i \rho_i A_{i+1}^\dagger) & \forall i \\
& \rho_0 = |0\rangle\langle 0|_{A\otimes M} & \\
& \rho_i \geq 0 & \forall i
\end{aligned}$$

Here ρ_i represents the state of Alice's and the message's registers, right before Alice performs her i-th action (see Figure 8.1). The dual of this SDP is

$$\begin{aligned}
(\text{SDP}_B,\ \text{dual}) \qquad & \min \quad \langle 0|Z_A(0)|0\rangle & \\
& Z_A(i) \otimes \mathbb{I}_M \geq A_{i+1}^\dagger (Z_A(i+1) \otimes \mathbb{I}_M) A_{i+1} & \forall i \\
& Z_A(n) = \pi_A & \\
& Z_A(i) = (Z_A(i))^\dagger & \forall i
\end{aligned}$$

Let $|\Psi_\ell\rangle$ be the state of the whole system at the ℓ-th round, assuming honest play. Then the analogue of (8.1) is

$$F_\ell = \langle \Psi_\ell | Z_A(\ell) \otimes \mathbb{I}_M \otimes Z_B(\ell) | \Psi_\ell \rangle \tag{8.2}$$

Strong duality then implies the condition $F_0 = p_{1*}p_{*0}$, while $F_n = 1/2$. The relation $F_\ell \geq F_{\ell+1}$ follows from the dual constraints, and we are done.

Acknowledgments

 The lecture notes are written by Nelly Ng, Thomas Vidick and Stephanie Wehner. We thank David Elkouss, Kenneth Goodenough, Jonas Helsen, Jérémy Ribeiro, and Andrea Coladangelo for proofreading.

Bibliography

[Aha+00] Dorit Aharonov et al. “Quantum bit escrow”. In: *Proceedings of the thirty-second annual ACM symposium on Theory of computing*. ACM. 2000, pages 705–714 (cited on page 11).

[Aha+16] Dorit Aharonov et al. “A Simpler Proof of the Existence of Quantum Weak Coin Flipping with Arbitrarily Small Bias”. In: *SIAM Journal on Computing* 45.3 (2016), pages 633–679 (cited on page 12).

[Amb+04] Andris Ambainis et al. “Multiparty quantum coin flipping”. In: *Computational Complexity, 2004. Proceedings. 19th IEEE Annual Conference on*. IEEE. 2004, pages 250–259 (cited on page 12).

[Amb01] Andris Ambainis. “A new protocol and lower bounds for quantum coin flipping”. In: *Proceedings of the thirty-third annual ACM symposium on Theory of computing*. ACM. 2001, pages 134–142 (cited on page 11).

[Ben+91] Charles H Bennett et al. “Practical quantum oblivious transfer”. In: *Annual International Cryptology Conference*. Springer. 1991, pages 351–366 (cited on page 6).

[Blu83] Manuel Blum. “Coin flipping by telephone a protocol for solving impossible problems”. In: *ACM SIGACT News* 15.1 (1983), pages 23–27 (cited on page 10).

[Bra+93] Gilles Brassard et al. “A quantum bit commitment scheme provably unbreakable by both parties”. In: *Foundations of Computer Science, 1993. Proceedings., 34th Annual Symposium on*. IEEE. 1993, pages 362–371 (cited on page 7).

[CK09] André Chailloux and Iordanis Kerenidis. “Optimal quantum strong coin flipping”. In: *Foundations of Computer Science, 2009. FOCS’09. 50th Annual IEEE Symposium on*. IEEE. 2009, pages 527–533 (cited on page 12).

[FS09] Serge Fehr and Christian Schaffner. “Composing quantum protocols in a classical environment”. In: *Theory of Cryptography Conference*. Springer. 2009, pages 350–367 (cited on page 4).

[Gol05] Oded Goldreich. *Foundations of cryptography: a primer*. Volume 1. Now Publishers Inc, 2005 (cited on page 4).

[GW07] Gus Gutoski and John Watrous. "Toward a general theory of quantum games". In: *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*. ACM. 2007, pages 565–574 (cited on page 12).

[LC97] Hoi-Kwong Lo and Hoi Fung Chau. "Is quantum bit commitment really possible?" In: *Physical Review Letters* 78.17 (1997), page 3410 (cited on page 9).

[LP09] Yehuda Lindell and Benny Pinkas. "Secure multiparty computation for privacy-preserving data mining". In: *Journal of Privacy and Confidentiality* 1.1 (2009), page 5 (cited on page 4).

[May97] Dominic Mayers. "Unconditionally secure quantum bit commitment is impossible". In: *Physical review letters* 78.17 (1997), page 3414 (cited on page 9).

[Moc07] Carlos Mochon. "Quantum weak coin flipping with arbitrarily small bias". In: *arXiv preprint arXiv:0711.4114* (2007) (cited on page 12).

[PS10] Rafael Pass and Abhi Shelat. *A Course in Cryptography*. Lecture notes available at `http://www.cs.cornell.edu/courses/cs4830/2010fa/lecnotes.pdf`. 2010 (cited on pages 6, 10).

[Unr10] Dominique Unruh. "Universally composable quantum multi-party computation". In: *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer. 2010, pages 486–505 (cited on pages 4, 8).

[Wat06] John Watrous. *Impossibility of quantum bit commitment*. Lecture notes from the Winter 2006 course "Introduction to Quantum Computing". 2006 (cited on page 9).

Lecture Notes

Quantum Cryptography Week 9:

Perfect security from physical assumptions

Contents

9.1 The noisy storage model

As we saw in the lecture notes of Week 8, security for two-party cryptography is hard to achieve. Even when given the ability to use quantum communication, we still cannot hope to achieve the security conditions. Yet, you maybe see that these are important problems that we need to solve on an everyday basis! One is therefore willing to make some reasonable assumptions on how powerful the dishonest party (adversary) can be, in order to obtain security guarantees. Security will thus hold as long as these assumptions are satisfied.

Classically, one typically makes so-called computational assumptions. This means that we assume that a particular problem, like factoring a large integer, requires a large amount of computational resources, and furthermore that the adversary has a relatively limited amount of computational resources - namely, insufficient to solve the difficult problem within a reasonable time frame. Once we build a quantum computer, however, many of these assumptions will no longer be valid! For example, we know that the Shor algorithm provides a much more efficient way to factor large numbers on a quantum computer! What's more, however, security can be broken retroactively: that is, if we build a quantum computer tomorrow, most two-party protocols that have been executed to date will lose their security/secrecy, since the adversary can now use the quantum computer to break the protocol. Clearly, this is very undesirable.

One way out of this dilemma is to make physical rather than computational assumptions. These assumptions only need to be valid during the execution of the protocol, but become irrelevant afterwards. Here, we will take a very simple assumption, namely we assume that the adversary can only store q qubits during one particular point during the protocol. Otherwise, the adversary remains all powerful: he/she may perform arbitrary quantum operations/computations, arbitrary encoding and decoding procedures, and store an infinite amount of classical information. This assumption is referred to as the bounded storage model. More generally, one can also invoke the noisy storage model, where the storage is not only bounded, but also noisy in general. In such cases, upper bounds quantities such as the entanglement assisted quantum capacity of the memory [Ber+13] leads to security. Given that even the most sophisticated experimental realization of quantum memories to date can reliably store no more than a few qubits for a few milliseconds [Bao+12; Cha+05; Rei+10], this is a technologically well motivated assumption. Before, or after the execution of the protocol, however, the attacker is allowed to have an arbitrary quantum memory, and even a quantum computer. In particular, this means that if tomorrow we can build better quantum memories, security can nevertheless never be broken retroactively.

In the next section, we have a look at a very simple protocol for this task.

9.2 1-2 Oblivious Transfer in the noisy-storage model

We have encountered the oblivious transferm in Week 8, and have seen that this is an important protocol, in the sense that any two-party protocol may be achieved by a combination of 1-2 oblivious transfer protocols, making it a fundamental building block of interest. We have also seen that despite the attempt to construct OT protocols that make use of quantum communication, no secure protocols exist – a malicious Bob could easily break the protocol by using for example a large quantum memory. Fortunately, this is extremely difficult to do in current-day technology, and therefore one may, by using the noisy storage model assumption, prove security for any Bob that has a limited quantum memory.

Protocol 1 Protocol for 1-2 OT in the noisy-storage model. Alice has inputs $s_0, s_1 \in \{0,1\}^\ell$, Bob has input $y \in \{0,1\}$.

1. Alice chooses a random strings $x = x_1, \ldots, x_n \in \{0,1\}^n$ and random basis $\theta = \theta_1, \ldots, \theta_n \in \{0,1\}^n$. She sends the bits encoded in the randomly chosen BB84 bases $H^{\theta_1}|x_1\rangle \otimes \ldots \otimes$

$H^{\theta_n}|x_n\rangle$ to Bob.

2. If $y = 0$, then Bob measures all of the qubits in the standard basis. If $y = 1$, he measures in the Hadamard basis. He records the resulting outcome string $\tilde{x} = \tilde{x}_1, \ldots, \tilde{x}_n$.
3. Both parties wait time Δt. (Storage assumption is applied!)
4. Alice sends to Bob the string $\theta_1, \ldots, \theta_n$. Bob computes the set of indicdes $\mathscr{I} = \{j \mid \theta_j = y\}$ where the measured in the same basis than Alice.
5. Alice chooses two random extractor functions as specified by random seeds r_0 and r_1. She computes $k_0 = Ext(x_+, r_0)$ and $k_1 = Ext(x_\times, r_1)$. Where x_+ is the substring of x where Alice encoded in the standard basis, and $x_\times$ is the substring where she used the Hadamard basis. She sends r_0 and r_1 to Bob.
6. Alice sends to Bob $m_0 = s_0 \oplus k_0$ and $m_1 = s_1 \oplus k_1$, where $\oplus$ denotes the bit wise xor.
7. Bob computes $k = Ext(x_{\mathscr{I}}, r_y)$ and $s_y = k_y \oplus r_y$.

Why does this protocol work? Let us first check that the protocol is correct, that is, Bob actually obtains s_y! Note that if there is no noise, then whenever $\theta_j = y$, we have $x_j = \tilde{x}_j$. That is, whenever Alice had encoded in the basis in which Bob measures, then Bob learns the corresponding element of Alice's bit string. This means that if Alice's now applies an extractor to hash down the elements of the strings corresponding to the standard and Hadamard basis respectively, then Bob knows one of them. Since Alice also sends him r_0 and r_1, he hence learns k_y, which acts like a key that encrypts s_y using one-time pad encryption, allowing him to recover s_y. Similar to the case of QKD, when a small amount of errors occur on the channel, information reconciliation can be performed in order to ensure that Bob is able to correct for the errors in $\tilde{x}$.

9.3 Security from quantum uncertainty

Is Protocol 1 secure? Let us first check security against dishonest Alice. Here, we want to show that Alice cannot learn y. If you look at the protocol above, it is clear that this is definitely the case: Bob never sends any information at all to Alice, from which we could learn anything about y!

The only difficulty is thus to show security against dishonest Bob. Here, we want to show that while Bob might learn one of the two strings, there is always a string $s_{\bar{c}}$ about which he knows (almost) nothing. As you can see from a protocol above, we have used our favorite trick encountered already in QKD, namely privacy amplification/randomness extraction. This means that if we could only ensure somehow that Bob's min-entropy about either x_+ or $x_\times$ is high, then by the properties of privacy amplification we could be sure that he knows nothing about the extracted key.

Before we do this, let us first consider whether we can say anything at all about Bob's min-entropy about *the entire* string x. To reason about this, let us first observe that something magic needs to happen in the waiting time Δt. Clearly, if Bob could store all of Alice's qubits, then he can just measure the entire string in the correct basis and learn the whole string x. This means that security against Bob can never be achieved if the number q of qubits that Bob can store is $q \geq n$.

Let us thus assume that $q < n$. Note that since we allow Bob to have an arbitrary quantum memory and computer *before* the waiting time, he can first store all of Alice's qubits, and perform an arbitrary quantum operation on all of them. For example, he might measure some of them, resulting in some classical information K. However, he can then keep only q qubits in his quantum register which we denote by Q. If we are interested in Bob's min-entropy about the entire string, we thus want to bound

$$H_{\min}(X_+X_\times|\Theta, K, Q) \tag{9.1}$$

where we have written $X_+X_\times$ for the string X to remind ourselves that we are ultimately interested in the two portions corresponding to the two different bases. To make his guess, Bob can use the

classical information K, the quantum register Q, as well as the basis information Θ that Alice sends to him after the waiting time.

In QKD, we saw that is it often much easier to show security against an adversary who is purely classical, so let us try and get rid of Q. As we have done so before in previous weeks, we can apply the chain rule for the min-entropy. Recall that the chain rule says

$$H_{\min}(X_+X_\times|\Theta,K,Q) \geq H_{\min}(X_+X_\times|\Theta,K) - \log|Q| \,, \tag{9.2}$$

$$= H_{\min}(X_+X_\times|\Theta,K) - q \,. \tag{9.3}$$

Hence, we could worry about a Bob who has only Θ and K. How could we possibly analyze this?

Again, let us think back to the tricks learned in QKD! By the guessing game, we could again think of Bob preparing qubits and send them to Alice. Alice chooses one of two random basis, after which she announces the basis choice to Bob. That is, we can use precisely the same guessing game that we had used in QKD to analzye the case of an eavesdropper Eve who has only classical information K! This gives the familiar

$$H_{\min}(X_+X_\times|\Theta,K) = n\left[-\log\left(\frac{1}{2}+\frac{1}{2\sqrt{2}}\right)\right] \approx 0.22n \,. \tag{9.4}$$

Of course, what we really want is to make a statement about the different parts X_+ and $X_\times$. That is, we want that there exists a $\bar{c}\in\{+,\times\}$ such that Bob's entropy about $X_{\bar{c}}$ is high. Fortunately, there is a beautiful lemma known as the *min-entropy splitting lemma* proven by Wullschleger which says that there exists some register $\bar{C}$ such that

$$H_{\min}(X_{\bar{C}}|\Theta,K,\bar{C}) \geq \frac{H_{\min}(X_+X_\times|\Theta,K)}{2} - 1 \,. \tag{9.5}$$

It is noteworthy that min-entropy splitting only holds if K really is classical which is why we first have to get rid of q. We thus know that Bob's min-entropy about at least one of the strings is large! Putting the two ideas together we thus have

$$H_{\min}(X_{\bar{C}}|\Theta,K,\bar{C}Q) \geq H_{\min}(X_{\bar{C}}|\Theta,K,\bar{C}) - q \tag{9.6}$$

$$\geq \frac{H_{\min}(X_+X_\times|\Theta,K)}{2} - 1 - q \,. \tag{9.7}$$

As usual we can now employ privacy amplifcation to say that Bob is ε-close to being ignorant, whenever

$$\ell < H_{\min}(X_{\bar{C}}|\Theta,K,\bar{C},Q) - O(\log 1/\varepsilon) - 1 \tag{9.8}$$

$$\approx 0.11n - q - O(\log 1/\varepsilon) - 2 \,. \tag{9.9}$$

This means that whenever $q \lesssim 0.11n$ we can have security for some $\ell > 0$! Or, reading it the other way around, assuming a maximum q for the adversary tells us that we need to send at least $n \approx 1/0.11q$ qubits in order to achieve security. By much more sophisticated analysis, it is now possible to show that security can be achieved as long as $q \leq n - O(\log^2 n)$ which is essentially optimal. We thus see that security is possible by sending just a few more qubits than Bob can store.

Acknowledgments

 The lecture notes are written by Nelly Ng and Stephanie Wehner. We thank David Elkouss, Kenneth Goodenough, Jonas Helsen, Jérémy Ribeiro, and Jalex Stark for proofreading.

Bibliography

[Bao+12] Xiao-Hui Bao et al. “Efficient and long-lived quantum memory with cold atoms inside a ring cavity”. In: *Nature Physics* 8.7 (2012), pages 517–521 (cited on page 3).

[Ber+13] Mario Berta et al. “Entanglement cost of quantum channels”. In: *IEEE Transactions on Information Theory* 59.10 (2013), pages 6779–6795 (cited on page 3).

[Cha+05] T Chaneliere et al. “Storage and retrieval of single photons transmitted between remote quantum memories”. In: *Nature* 438.7069 (2005), pages 833–836 (cited on page 3).

[Rei+10] KF Reim et al. “Towards high-speed optical quantum memories”. In: *Nature Photonics* 4.4 (2010), pages 218–221 (cited on page 3).

Lecture Notes

Quantum Cryptography Week 10: Further topics

Contents

10.1 Delegated computation

As powerful as quantum computers may be, in the foreseeable future they are likely to remain rather bulky machines dedicated to special-purpose computations. One of the first to have realized the potential of quantum mechanics for computation is Richard Feynman, who thought that one of the main applications of quantum computing devices would be precisely to the problem of *simulation*: quantum computers to simulate quantum systems [Fey82]! Indeed predicting the properties of physical materials such as superconductors, or certain chemical molecules, are some of the most challenging problems faced by experimentalists, as running accurate simulations of such systems on a classical computer is extremely costly, if at all possible.

Today we see Feynman's idea approaching realization. The largest quantum computers to date number in the hundreds of qubits, but these computers can only perform specialized tasks: for instance, the qubits might be arranged on a 2D grid such that one is only able to apply certain gates on neighboring qubits. Nevertheless, such large bulky machines are in principle universal, meaning that any quantum computation can be translated into one that can be executed on the machine, with a possibly rather large (but nevertheless polynomial) blow-up.

This sets the stage for the scenario of delegated computation. Suppose a user Alice has a quantum circuit $\mathscr{C}$ in mind, that she would like to execute on some input x in order to learn the outcome $\mathscr{C}(x)$. All of Alice's data, x and $\mathscr{C}$, is classical: we can assume x is a classical bit string, and $\mathscr{C}$ is specified by a sequence of two-qubit gates. We can also assume the outcome is classical: let's say it is the outcome of a standard basis measurement performed on a specially designated output qubit of $\mathscr{C}$.

Now, unfortunately Alice herself does not have a universal quantum computer! Maybe she has a tiny desktop machine that lets her play around with BB'84-like operations: prepare or measure single qubits, possibly store a couple qubits at a time in memory, but no more. Luckily Alice has the possibility of buying computation time on a quantum server, with which she could interact over the internet, or maybe even over a simple BB'84-type quantum communication channel that allows the exchange of one qubit at a time. So Alice could send x and the description of $\mathscr{C}$ to the server, who would perform the computation and return the outcome — right?

Remember that this is a crypto class. Alice might not trust this server. For one she'd like to have a way to verify that the outcome provided to her is correct. What if the server is lazy and systematically claims the outcome of her computation, $\mathscr{C}(x)$, equals '0'? Since Alice has no quantum computer herself she has no means of checking the outcome! A second property Alice could require is that the computation be private: while she certainly wants to learn $\mathscr{C}(x)$, she'd rather not have the server know that she is interested in circuit $\mathscr{C}$, or in input x, as these might contain private data.

Let's restate these conditions a bit more formally as the requirements that the computation be *correct*, *verifiable* and *blind*.

Definition 10.1.1 — Delegated computation. In the task of delegated computation, a user Alice (sometimes called the *verifier*) has an input $(x,\mathscr{C})$, where x is a classical string and $\mathscr{C}$ the classical description of a quantum circuit. Alice has a multiple-round interaction with a quantum server (sometimes also called *prover*). At the end of the interaction, Alice either returns a classical output y, or she aborts. A protocol for delegated computation is called:

- *Correct* if whenever both Alice and the server follow the protocol, with high probability Alice accepts (she does not abort) and $y = \mathscr{C}(x)$.
- *Verifiable* if for any server deviating from the protocol, Alice either aborts or returns $y = \mathscr{C}(x)$.
- *Blind* if for any server deviating from the protocol, at the end of the protocol the server has no information at all about Alice's input $(x,\mathscr{C})$.

The properties of verifiability and blindness are stated rather informally. A precise definition satisfying all the desired properties (universal composability in particular) would take us many pages. Such a definition was given using the framework of *abstract cryptography* in [Dun+14].

The informal definition given above will be sufficient for our purposes. Note that in spite of being rather similar neither of the properties of verifiability or blindness directly implies the other. In practice it will often be the case that verifiability follows from blindness by arguing, using "traps", that if a protocol is already blind, the server's trustworthiness can be tested by making it run "dummy" computations for which Alice already knows the output, without the server being able to distinguish whether it is asked to do a real or dummy computation. We will see an example of this technique later on.

What are good protocols for delegating quantum computations? It turns out that we don't have a fully satisfactory answer yet: this is an active area of research! Many interesting ideas are being pursued, but none is perfect. In these notes we'll survey the three most prominent approaches. The first construction shows how arbitrary quantum circuits can be delegated, as long as the verifier has the ability to prepare certain specific single-qubit states and communicate them to the server. The second construction achieves a similar result, using a very different idea: *measurement-based* quantum computation. The third construction has a wholly different flavor. It achieves delegated computation by a purely classical verifier, with no quantum abilities whatsoever. However, the downside is that the verifier now has to interact with *two* isolated quantum servers. This method relies on similar techniques as we have seen in the analysis of device-independent QKD (in particular the use of the CHSH game for testing that the two servers share EPR pairs).

10.1.1 Preliminaries on efficient quantum computation

Before we delegate quantum computations, let's first review briefly what a quantum computation *is*. As hinted at earlier, there are many distinct models for quantum computation, each capable of universal computation (and thus of simulating any other). We'll see three of them in these notes.

The most natural model is called the quantum circuit model. Here a computation is represented by the action of a circuit $\mathscr{C}$ on n input qubits. The input qubits are initialized in an arbitrary quantum state that contains the input of the computation; we will mostly be concerned with classical inputs of the form $|x\rangle$, where $x \in \{0,1\}^n$, but quantum inputs can also be considered. The circuit $\mathscr{C}$ itself is specified by a list of m gates, which are one- or two-qubit unitaries that act on a subset of the qubits. For instance, a simple 4-qubit circuit could be specified as the ordered list $((H,1),(\text{CNOT},(2,3)),(H,3),(Z,4))$. It is then usually assumed that once all gates have been applied the first qubit is measured in the standard basis to produce a single bit representing the classical outcome of the circuit (though quantum outputs can be considered as well). We will write $\mathscr{C}(x)$ for the outcome of a measurement of the output qubit of $\mathscr{C}$ in the standard basis, when $\mathscr{C}$ is executed on the state $|x\rangle$. Thus $\mathscr{C}(x)$ should be treated as a random variable; for simplicity we'll assume that for each possible input x it takes on a certain value with high probability, and we'll write $\mathscr{C}(x)$ for that fixed value.

The class BQP of "efficiently computable functions" in the quantum circuit model is formally defined as follows.[1]

Definition 10.1.2 — BQP. A family of functions $\{f_n : \{0,1\}^n \to \{0,1\}\}$ is in BQP if for every integer n there exists a circuit $\mathscr{C}_n$ whose description size (number of gates and precision of the entries in each gate) is polynomial in n and such that for every $x \in \{0,1\}^n$, a measurement of the output qubit of the execution of $\mathscr{C}_n$ on $|x\rangle$ returns $f(x)$ with probability at least $2/3$.

The definition of BQP allows arbitrary circuits $\mathscr{C}$, as long as they can be efficiently specified

[1] In complexity theory we often talk about "languages" rather than functions, were a language is a subset $L \subseteq \{0,1\}^*$. It is then natural to associate a Boolean function to any language, and vice-versa.

as a list of single- or two-qubit gates. An important theorem in quantum computing, the Solovay-Kitaev theorem, states that it is in fact possible to restrict the set of gates allowed to some simple sets of gates, called "universal gate sets": a gate set is universal if any circuit that has an efficient implementation, has an efficient implementation that uses only gates from that set. An example gate set that we will use later on is the set

$$\mathscr{G} = \left\{ G = \begin{pmatrix} \cos(\pi/8) & -\sin(\pi/8) \\ \sin(\pi/8) & \cos(\pi/8) \end{pmatrix}, \text{CNOT} \right\},$$

where G implements a $\pi/4$ rotation around the y axis of the Bloch sphere and CNOT a controlled-X operation (on any two qubits of the circuit). Another example of a popular gate set is the set

$$\mathscr{G}' = \left\{ H, T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}, \text{CNOT} \right\},$$

but there are many others; which gate set to use depends on the task at hand (as well as the architecture of the quantum computer, as certain physical implementations can implement certain gates much more easily than others).

We will use one more useful feature of quantum (as well as classical) circuits, which is the notion of a "universal" circuit. Due to the universality of quantum circuits for efficient quantum computation, and its equivalence with the model of quantum Turing machines, it is possible to show the following.

Theorem 10.1.1 — Universal circuit. For any integer n and size parameter s there exists a fixed circuit $\mathscr{C}_U$ acting on $n+m$ qubits, where m is a polynomially bounded function of n and s, such that for any circuit $\mathscr{C}$ of size at most s expressed using the gate set $\mathscr{G}$, and any input $x \in \{0,1\}^n$ to $\mathscr{C}$, there is a $z \in \{0,1\}^m$ (efficiently computable from $\mathscr{C}$) such that $\mathscr{C}_U(x,z)$ has the same distribution as $\mathscr{C}(x)$.

10.1.2 The Pauli group and Clifford gates

We've already encountered the 4 single-qubit Pauli matrices, $\mathscr{P} = \{I, X, Y, Z\}$. These form a group, in the sense that for any two Pauli matrices P and Q, the product PQ is again a Pauli matrix. A single-qubit *Clifford gate* U is any operation that preserves the Pauli group, in the sense that $UPU^\dagger$ is a Pauli matrix for any Pauli P. If U is a two-qubit gate, we similarly require that $UPU^\dagger \in \pm\mathscr{P}^{\otimes 2}$ for any $P \in \mathscr{P}^{\otimes 2}$.

Exercise 10.1.1 Clearly the Pauli matrices are Clifford gates. Show that the Hadamard, phase $P = T^2$ and CNOT gates are Clifford gates. Do you see other examples? Is the G gate considered above a Clifford gate? How about the T gate? ∎

The defining property of Clifford gates is very useful, and plays an important role in delegated computation — we will see why. Unfortunately it turns out that there is no universal gate set made only of Clifford gates: any universal set of gates for quantum computation must include at least one non-Clifford gate. This will be a source of many headaches when trying to implement delegated computation.

10.2 Verifiable delegation of quantum circuits

Our first approach to delegated computation is based on the idea of *computing on encrypted data*. Recall the quantum one-time pad from Week 1. Suppose we have an n-qubit density matrix ρ. To encrypt it using the one-time pad we select two n-bit strings $a, b \in \{0,1\}^n$ uniformly at random, and

return $\tilde{\rho} = X^a Z^b \rho (X^a Z^b)^\dagger$, where X^a denotes applying a Pauli X operator on all qubits i such that $a_i = 1$; similarly for Z^b with Pauli Z operators. If ρ represents the input x to the circuit, $\rho = |x\rangle\langle x|$, then the result of applying the quantum one-time pad is still classical, $\tilde{\rho} = |x \oplus a\rangle\langle x \oplus a|$. So this is an operation the client Alice can easily perform by herself, and send $\tilde{\rho}$ to the server. If she keeps a local copy of the strings a, b but does not communicate them to the server her input x remains perfectly private.

Now let's see how Alice could make the server execute a circuit $\mathscr{C}$ by acting directly on the encrypted state $\tilde{\rho}$. The goal is to find a transformation $\tilde{\mathscr{C}}$ for the server to apply, such that $\tilde{\mathscr{C}}(\tilde{\rho}) = \widetilde{\mathscr{C}(\rho)}$, an encrypted version of $\mathscr{C}(\rho)$ from which Alice should be able to recover the output $\mathscr{C}(x)$.

The circuit $\mathscr{C}$ can be expressed using gates taken from a universal gate set, for example the set $\mathscr{G}' = \{H, \text{CNOT}, T\}$ considered in Section 10.1.1. Even though it is not needed, to warm up let's assume we'd also allow ourselves to use X gates, and that the first gate in $\mathscr{C}$ is an X to be applied on the first qubit. We'd like the server to evaluate

$$X^a Z^b (X \rho X^\dagger)(X^a Z^b)^\dagger = X^{a \oplus e_1} Z^b \rho (X^{a \oplus e_1} Z^b)^\dagger = X \tilde{\rho} X^\dagger,$$

where e_1 is the bit string with a single 1 in the first position. These equations show that the server doesn't even need to apply the X gate: it is sufficient for Alice to update her one-time pad key from (a, b) to $(a \oplus e_1, b)$. So the X gate is easy. You can see that any Pauli gate, single- or multi-qubit, will be similarly easy.

Let's move one step further and consider the implementation of a Clifford gate — let's take the example of a Hadamard gate on the second qubit, H^{e_2}. Using $HXH = Z$, observe that

$$(X^a Z^b)(H^{e_2} \rho (H^{e_2})^\dagger)(X^a Z^b)^\dagger = (-1)^{a_2 b_2} H^{e_2} X^{a'} Z^{b'} \rho (X^{a'} Z^{b'})^\dagger,$$

where (a', b') is obtained from (a, b) by exchanging the bits a_2 and b_2. Hence if Alice instructs the server to apply an H gate on the second *encrypted* qubit, the effect is the same as if the server had applied the H gate directly on the second *unencrypted* qubit — as long as she updates her one-time pad key (a, b) to (a', b') as described above. The following exercise asks you to show that a similar trick can be employed for any Clifford gate.

Exercise 10.2.1 Let U be any one- or two-qubit Clifford gate. Show that the effect of applying U to the encrypted state $\tilde{\rho}$ is equivalent to the application of U on ρ, up to an update rule on the one-time pad key (a, b). Work out the update rule in the case of the phase and CNOT gates. ■

So Alice can orchestrate the whole computation within the server, only having to keep track of simple updates on her one-time pad keys? Unfortunately, remember from Section **??** that no set of Clifford gates is universal — we need to show how to implement one more gate, for instance the T gate considered in the universal set $\mathscr{G}'$. Because the T gate is non-Clifford, applying it to the encrypted state $\tilde{\rho}$ will have a more complicated effect, that we can't keep track of by a simple modification of the one-time pad keys. Instead, we'll show how Alice can make the server implement a T gate on the encrypted state by using the idea of *magic states*.

10.2.1 Computation with magic states

The idea for magic states is that the computation of certain complicated gates on an arbitrary state can be replaced by a simple computation using certain ancilla states called "magic states". Let's see this for the T gate, the only gate that we still need to figure out how to implement. The magic state we need is

$$|\pi/4\rangle = T|+\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{e^{i\pi/4}}{\sqrt{2}}|1\rangle. \tag{10.1}$$

Preparing this state itself requires applying a T gate. But the point is that we only need to apply the gate to a fixed, known input state, that is independent of the state $|\psi\rangle$ on which we really want to apply the T gate. So the preparation of single-qubit magic states is a relatively simple task that we could ask Alice to perform, if she had access to a small, basic single-qubit quantum computer.

Suppose we are given a single-qubit state $|\psi\rangle$ on a register A_1, and initialize an ancilla register A_2 in the $|\pi/4\rangle$ state. Consider the following circuit: first, apply a CNOT, controlled on A_2 and acting on A_1. Second, measure register A_1 in the computational basis, obtaining an outcome c. Third, apply a gate P^c to register A_2 (see Figure 10.1). What is the corresponding post-measurement state of register A_2? It is a good exercise to verify that this is $X^c Z^c T|\psi\rangle$. That is, up to a simple (XZ) correction, we applied the T gate to $|\psi\rangle$ while only requiring a magic state, a CNOT, a measurement in the computational basis, and a controlled-P gate.

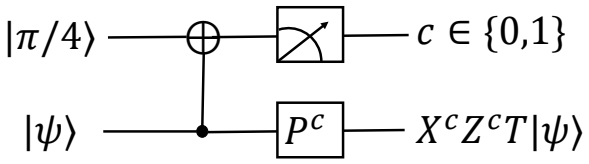

Figure 10.1: Teleporting into a T gate. The state $|\pi/4\rangle$ is defined in (10.1).

Exercise 10.2.2 Suppose that instead of being applied directly to the state $|\psi\rangle$, the circuit described above is applied to an encrypted version of $|\psi\rangle$, $X^a Z^b|\psi\rangle$. Show that the outcome of the circuit is then $P^a X^{a'} Z^{a'} T|\psi\rangle$, for some bits a' and b' depending on a, b and c. (Convince yourself that the same calculation works out in case $|\psi\rangle$ is not pure, but a reduced density ρ on a single qubit.) ■

Note the phase gate that we picked up in the exercise. This also needs to be corrected. But the phase is applied on the encrypted state. Alice could instruct the server to apply P^a, but this would require revealing the bit a, which is part of the key. There is a way around this that involves adding a little bit of randomization in the choice of magic state we use (essentially, considering a one-time padded magic state), so as to guarantee that the phase correction to be applied is always independent of the one-time pad key. You can try to work this out as an exercise, or look at the paper [Bro15] for a possible solution.

10.2.2 Blindness

With the main ingredients in place, there are still a number of difficulties we need to overcome. Let's first consider the blindness property. For this, we assume the server is completely honest, and we're only worried that it may be able to obtain information about Alice's input x or circuit $\mathscr{C}$, while still following the correct protocol.

Unfortunately it seems clear that the circuit itself has to be revealed to the server: in the scheme we described in the previous section each application of a T gate will involve an interaction between the client and the server. But there is a simple way out: Alice can instruct the server to execute a fixed "universal" circuit (as in Theorem 10.1.1), and instead encode the actual circuit she is interested in as part of the input. (Note that this method is not very practical, as there is a large computational overhead in using such a universal circuit.)

So now the only thing that Alice needs to keep hidden is her new input (x, z), and this is precisely what the one-time pad achieves. But we should be careful: even though the OTP certainly hides the input at the start of the interaction, subsequent interactions could in principle reveal more

information. The only gate which involves an interaction between Alice and the server is the T gate, which according to the calculation performed in Exercise 10.2.2 requires Alice to send the server some classical information in order for it to determine whether a P correction should be applied or not (see Figure 10.1). As mentioned earlier it is possible to choose a magic state uniformly at random from a set of 8 possibilities, in a way that both the magic state and the correction bit will appear uniformly random to the server, irrespective of the measurement outcome c it reports to Alice. Once again we refer you to [Bro15] for the details on how this can be performed.

10.2.3 Verifiability

The protocol we described thus satisfies the blindness property (provided we use a universal circuit and implement the T gates as sketched above), but so far it is not verifiable: the client has no guarantee that the server performs the required computation; indeed, no check is performed and the client would accept any answer (note that under the OTP, any string can be interpreted as a valid ciphertext).

The idea to enforce verifiability is to combine the protocol with some "test runs". The original protocol is now called a "computation run". In contrast, in a test run the client knows what the outcome should be, and she will check that the server returns the correct value. But the server will not be able to distinguish test runs from computation runs, and as a consequence we'll have the guarantee that the server is also being honest in a computation run.

There are two types of test runs, X-test and Z-test. In an X-test run, the computation is executed on an encryption of the all-0 input $|0\rangle^{\otimes n}$. In a Z-test run, the same computation is executed on an encryption of $|+\rangle^{\otimes n}$. The main trick to ensure that the verifier is able to keep track of the computation is that all gates in a test run are replaced by *identity* gates, without the prover noticing! Note that we already know how to do this for Pauli gates, as these do not involve the server anyways (the verifier only has to update her one-time pad keys). The H gate requires a bit more work, but the idea is simple: since an H exchanges the standard basis and the Hadamard basis we can think of it as exchanging between an X-test run to an Z-test run, so that the verifier can still perfectly keep track of the state that the circuit should be in. The T gate, of course, is the interesting one. The idea is to modify the implementation described in Section 10.2.1 by changing the magic state, as well as the update rule, in a way that is un-noticeable by the server but will result in an application of the identity gate instead of the T. The following exercise asks you to work out how this can be done.

Exercise 10.2.3 Consider the following procedure for implementing a T gate on the single-qubit state $|\psi\rangle_{A_1}$ using a magic state in register A_2. The client first selects two bits $d, y \in \{0,1\}$ uniformly at random, and prepares the magic state $Z^d P^y T|+\rangle_{A_2}$, where $P = T^2$ is the phase gate. The client sends system A_2 to the server, who performs a CNOT, controlled on A_1 and with target A_2. The server measures A_1 in the computational basis, obtaining an outcome $c \in \{0,1\}$ that it sends to the client. The client then sends back $x = y \oplus c$ to the server, who applies a gate P^x to the remaining system A_2.

1. Show that the state of A_2 at the end of this procedure is $X^c Z^{c(y\oplus 1)\oplus d\oplus y} T|\psi\rangle$, i.e. it is an encryption (using a key known to the client) of $T|\psi\rangle$.

Next let's suppose we're doing a computation run, so that $|\psi\rangle = X^a|0\rangle_{A_1}$ for some $a \in \{0,1\}$. The client would like to perform the identity instead of a T gate, without the server noticing. This can be done by executing precisely the same circuit, except the magic state is replaced by $X^d|0\rangle_{A_2}$ (it does not depend on y).

2. Show that with the magic state replaced by $X^d|0\rangle_{A_2}$ the interaction results in a register A_2 in state $X^d|0\rangle_{A_2}$. Show that in this case the outcome c of the server's measurement is deterministically related to a and d in a simple way.
3. Can you find a similar modification, with a different magic state, that will implement the

identity for the case of a Z-test run, where $|\psi\rangle_{A_1} = Z^b|+\rangle_{A_1}$ for some $b \in \{0,1\}$? ■

The exercise shows that simply by changing the magic state used in the implementation of the T gate, the client can force that gate to act as identity in an X- or Z-test run. Moreover, due to the random bits d, y used in the preparation of the magic state you can verify that, from the point of view of the server, these magic states look uniformly distributed, and thus it has no way of telling which "gadget" — for a T gate or the identity — it is really implementing.

In a test run the verifier knows exactly what the outcome of the circuit should be, so that it can verify the answer provided by the client. Is this enough to ensure that the server cannot cheat in a computation run? After all, we can imagine that the server may be able to perform certain attacks that do not affect simple computations, where the state is always a tensor product of single qubits encoded in the computational or Hadamard bases, but such that the attack would perturb the kind of highly entangled states that will show up at intermediate stages in a more complex circuit.

To show that this is not the case — that any significant attack will necessarily have a noticeable effect on either the X- or Z-test runs, the idea is to use an observation called the "Pauli twirl", that you are asked to work out in the next exercise.

Exercise 10.2.4 — Pauli twirl. Let ρ be a single-qubit density matrix, and $P, P' \in \mathscr{P}$, where $\mathscr{P} = \{I, X, Y, Z\}$ is the set of single-qubit Pauli operators. Show that $\sum_{Q\in\mathscr{P}} (Q^\dagger P Q)\rho(Q^\dagger (P')^\dagger Q)$ equals $P\rho P^\dagger$ if $P = P'$, and is 0 otherwise. Show that the same result holds for n-qubit Pauli operators. ■

The Pauli twirl allows us to argue that, thanks to the use of the quantum one-time pad, any "attack" of the server boils down to the application of a Pauli operator at the last step of the circuit. Indeed, suppose first that the interaction performed between the client and the server results in the correct circuit $\mathscr{C}$ being implemented, except at the last step the server applies an arbitrary "deviating unitary" U. Thus the outcome is $U\tilde{C}\tilde{\rho}\tilde{C}^\dagger U^\dagger$, where $\tilde{C}$ is the unitary Alice instructed the server to implement, and $\tilde{\rho}$ the initial OTP-encoded state sent by the client. Due to the OTP, $\tilde{\rho}$ has the form $\tilde{\rho} = \sum_{Q\in\mathscr{P}} Q|x\rangle\langle x|Q^\dagger$, where $|x\rangle$ denotes the real input state that the client would like the computation to be performed on. Moreover, for any Q there is a correction $c(Q) \in \mathscr{P}$ applied by the client, which is such that $c(Q)\tilde{C}Q|x\rangle\langle x|Q^\dagger\tilde{C}^\dagger(c(Q))^\dagger = C|x\rangle\langle x|C^\dagger$. Thus, after applying $c(Q)$ to the corrupted circuit,

$$\begin{aligned}
&\sum_{Q\in\mathscr{P}} c(Q)U\tilde{C}Q|x\rangle\langle x|Q^\dagger\tilde{C}^\dagger U^\dagger(c(Q))^\dagger \\
&\quad= \sum_{Q\in\mathscr{P}} c(Q)U(c(Q))^\dagger c(Q)\tilde{C}Q|x\rangle\langle x|Q^\dagger\tilde{C}^\dagger(c(Q))^\dagger c(Q)U^\dagger(c(Q))^\dagger \\
&\quad= \sum_{Q\in\mathscr{P}} c(Q)U(c(Q))^\dagger C|x\rangle\langle x|C^\dagger c(Q)U^\dagger(c(Q))^\dagger \\
&\quad= \sum_{P\in\mathscr{P}} |\alpha_P|^2 PC|x\rangle\langle x|C^\dagger P^\dagger,
\end{aligned}$$

where for the last step we decomposed $U = \sum_{P\in\mathscr{P}} \alpha_P P$ in the Pauli basis, and used the property of the Pauli twirl proved in Exercise 10.2.4.

This computation shows that any unitary applied by a malicious server at the end of the honest circuit is equivalent to a convex combination of Pauli operators. But any such non-trivial operator will be detected in either the X- or Z-test runs, as it will result in one of the outcomes being flipped in either the standard or Hadamard bases.

To conclude we need to deal with the case where the server applies a deviating unitary, not at the end of the circuit, but at some intermediate step. But this case can be reduced to the former! Indeed, we can always think of a "purified" version of the whole protocol, where all measurements

are deferred until the end. Suppose the unitary $\tilde{C}$ the server is supposed to implement decomposes as $\tilde{C} = \tilde{C}_2\tilde{C}_1$, and that the server applies a deviating unitary U in-between the two circuits. The result can be written as

$$\tilde{C}_2 U \tilde{C}_1 = (\tilde{C}_2 U \tilde{C}_2^\dagger)\tilde{C}_2\tilde{C}_1 = (\tilde{C}_2 U \tilde{C}_2^\dagger)\tilde{C},$$

where we used that $\tilde{C}_2$ is unitary, and hence $\tilde{C}_2^\dagger\tilde{C}_2 = \mathbb{I}$. Thus the deviation U is equivalent to applying another deviating unitary $U' = \tilde{C}_2 U \tilde{C}_2^\dagger$ at the end of the circuit, and we are back to the analysis performed in the previous case: if the deviation has a non-trivial effect it will be detected by the client in one of the test runs.

10.3 Delegation in the measurement-based model

Our second scheme for delegated computation has a similar flavor to the one presented in the previous section, but at its heart it is based on a completely different approach to universal quantum computation. So far we have encountered the circuit model for performing quantum computations. From the point of view of computer science this is the most natural model, as it is a direct analogue of the classical circuit model on which the architecture of our (classical) computers is based. However, quantum information allows for other, much more exotic, models of computation. Many of these models were originally proposed with the idea that they might be more powerful than the circuit model, although ultimately they were proved equivalent. This includes the adiabatic model for computation [Aha+08] and the measurement-based model that we will discuss in this section.

The highlight of measurement-based quantum computation (MBQC) is that it allows one to implement an arbitrary quantum computation (specified by a circuit using some universal gate set) solely by executing an (adaptive) sequence of single-qubit measurements on a fixed, universal starting state. Seems impossible? Let's first give an overview of how this model works, and then we'll explain how MBQC can be used to achieve blind, verifiable delegated computation.

10.3.1 Measurement-based computation

Measurement-based computation is based on an idea very similar to *teleportation-based computation*, a model to which we return in the next section. It is the idea that a complete quantum computation, including the preparation of the initial state and the application of gates from a universal set, can be performed by making a sequence of adaptive measurements on a fixed universal state, simultaneously "teleporting" the input state from one qubit to the next while at the same time applying unitary transformations on the state.

Let's do a simple example first. Suppose we have a qubit initialized in the state $|\psi\rangle_A = \alpha|0\rangle_A + \beta|1\rangle_A$. Suppose a second qubit is created in the state $|+\rangle_B$, and a CTL-Z operation is performed, controlling on the first qubit to perform a phase flip on the second. Then the joint state of the system is $|\psi\rangle_{AB} = \alpha|0\rangle_A|+\rangle_B + \beta|1\rangle_A|-\rangle_B$. Suppose now we measure the first qubit in the Hadamard basis. What happens to the second qubit? Let's re-write

$$\begin{aligned}|\psi\rangle_{AB} &= \alpha|0\rangle_A|+\rangle_B + \beta|1\rangle_A|-\rangle_B \\ &= \frac{1}{\sqrt{2}}|+\rangle_A(\alpha|+\rangle_B + \beta|-\rangle_B) + \frac{1}{\sqrt{2}}|-\rangle_A(\alpha|+\rangle_B - \beta|-\rangle_B).\end{aligned}$$

The measurement rule states that if we get the outcome "+" the second qubit is projected to $|\psi'\rangle_B = \alpha|+\rangle_B + \beta|-\rangle_B$, and if we get a "−" it is projected to $\alpha|+\rangle_B - \beta|-\rangle_B$. In the first case, $|\psi'\rangle_B = H|\psi\rangle$, and in the second $|\psi'\rangle_B = XH|\psi\rangle$. More succinctly put, $|\psi'\rangle_B = X^m H|\psi\rangle$ where $m \in \{0,1\}$ denotes the outcome of the measurement: $m = 0$ in case of "+" and $m = 1$ in case of "−". Thus, up to a "Pauli correction" X^m, we managed to apply a Hadamard gate simply by making a single-qubit measurement on the appropriate state.

For an arbitrary $\phi \in [0, \pi/2)$ let

$$|+_\phi\rangle = \frac{1}{\sqrt{2}}|0\rangle + e^{i\phi}\frac{1}{\sqrt{2}}|1\rangle \qquad \text{and} \qquad |-_\phi\rangle = \frac{1}{\sqrt{2}}|0\rangle - e^{i\phi}\frac{1}{\sqrt{2}}|1\rangle, \tag{10.2}$$

and

$$U_z(\phi) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{pmatrix}, \qquad U_x(\phi) = HU_z(\phi)H.$$

The rotations $U_z(\phi)$ and $U_x(\phi)$ together generate a universal set of single-qubit gates, as any rotation on the Bloch sphere can be implemented as $U_z(\varphi_3)U_x(\varphi_2)U_z(\varphi_1)$ for an appropriate choice of φ_1, φ_2 and φ_3. The following exercise asks you to generalize the example of the Hadamard gate to any single-qubit rotation that can be decomposed in this way.

Exercise 10.3.1 Modify the method we described to apply a Hadamard gate by instead performing a measurement of the first qubit in the basis $\{|+_\varphi\rangle, |-_\varphi\rangle\}$. Show that the second qubit is then projected on the state $X^m HU_z(\varphi)|\psi\rangle$, where $m \in \{0,1\}$ indicates the measurement outcome.

Now consider a sequence of three measurements with angles φ_1, φ_2 and φ_3. That is, suppose a first qubit is in state $|\psi\rangle_A$, and three additional qubits are created in the $|+\rangle$ state and organized on a line. Three CTL-Z operations are performed from left to right. Then the first qubit is measured in basis $\{|+_{\varphi_1}\rangle, |-_{\varphi_1}\rangle\}$, obtaining an outcome $m_1 \in \{0,1\}$, the second qubit is measured with angle φ_2, obtaining outcome m_2, and finally the third qubit is measured with angle φ_3, obtaining outcome m_3. Show that the state of the fourth qubit can then be written as

$$|\psi'\rangle_D = X^{m_3}Z^{m_2}X^{m_1}HU_z((-1)^{m_2}\varphi_3)U_x((-1)^{m_1}\varphi_2)U_z(\varphi_1)|\psi\rangle. \tag{10.3}$$

[Hint: you may use the identities $XU_z(\phi) = U_z(-\phi)X$ and $HU_z(\phi)H = U_x(\phi)$, valid for any real ϕ.] ■

The exercise *almost* lets us apply an arbitrary rotation $U_z(\varphi_3)U_x(\varphi_2)U_z(\varphi_1)$, except there are these annoying "corrections", the X and Z operations and the Hadamard to the left, as well as extra $(-1)^{m_i}$ phases in the angles. But these can be dealt with easily! For the phases, note that we perform the measurements sequentially, and the phase flip that got applied to a certain angle only depends on the outcome of the measurement performed right before. For the case of the calculation performed in the exercise, if we *really* had wanted to end up with $U_x(\varphi_2)$, after having obtained outcome m_1 we could have updated our choice of angle in which to measure to $\varphi_2' = (-1)^{m_1}\varphi_2$. As for the X, Z and H corrections at the end of the computation, we can handle those at the time of final measurement: they correspond to corrections that will need to be applied once we measure the final qubit (this is similar to how we handled the one-time pad in the previous section).

This shows how any sequence of single-qubit rotations can be applied to a qubit. You would start with a line of m qubits, each initialized in the $|+\rangle$ state. Then apply CTL-Z operations on all pairs of neighboring qubits, from left to right. This corresponds to preparing a $1 \times m$-dimensional "brickwork state", a universal resource for single-qubit computation. Suppose for simplicity the initial qubit is meant to be initialized in the $|+\rangle$ state (if it is not you can modify the circuit so that the first gate applied prepares the correct qubit). Any rotation can be applied by decomposing it in the form $U_z(\varphi_3)U_x(\varphi_2)U_z(\varphi_1)$ and making the correct sequence of measurements on three qubits, keeping track of successive measurement outcomes to update the angles and the X, Z and H "corrections" that tag along to the left of the description of the state of the qubit, as in (10.3) (note that you do not need to remember all measurement outcomes, but only their combined effect in terms of a power of X and a power of Z).

What if we have a multi-qubit computation? We won't give the details, but the general idea is the same. Since we already know how to implement arbitrary single-qubit gates, to get a universal

gate set it suffices to implement a 2-qubit CNOT gate. We'll use multiple lines of qubits, one per qubit of the computation. The lines are connected by vertical CTL-Z operations once every three qubits (in a slightly shifted manner). A two-qubit CNOT gate can then be applied using similar ideas as we described, but performing measurements on the two lines associated with the two qubits on which the gate acts. We'll leave the details as an exercise, and refer you to the notes [BB06] for detailed explanations.

10.3.2 Blind delegation in the MBQC model

Now that we have seen how to perform an arbitrary computation in the MBQC model, let's see how the computation can be delegated to an untrustworthy server. Let's imagine that the client, Alice, has a sequence of single-qubit measurements, specified by angles $\{\varphi_{ij}\}_{1\leq i\leq n, 1\leq j\leq m}$ and update rules (depending on prior measurement outcomes), that she wishes to apply on an $n\times m$ brickwork state in order to implement a certain n-qubit quantum circuit she is interested in. Let's also assume for simplicity that the outcome of the last measurement would (possibly after a Pauli correction if needed) give her the answer she is looking for.

Of course Alice could tell the server to prepare the $n\times m$ brickwork state and then instruct it, through a classical interaction, to perform the measurements specified by the φ_{ij}. The server would report the outcomes, Alice would perform the updates, and tell the server the next angle to measure in. But clearly this would be neither blind nor verifiable.

The key idea is then for Alice to (partially) prepare some kind of "one-time padded" version of the brickwork state, on which the server will implement the computation without ever having any information about the "real" angles φ_{ij}.

Consider the following outline for a protocol.

Protocol 1 Fix a set of "hiding" angles $D=\{0,\pi/4,2\pi/4,\ldots,7\pi/4\}$.

1. For each of the nm qubits of the brickwork state, Alice chooses a random $\theta_{i,j}\in D$, prepares the state $|+_{\theta_{i,j}}\rangle$, and sends it to the server.
2. The server arranges all qubits it receives in the shape of an $n\times m$ brickwork state, and performs CTL-Z operations on neighboring qubits as required.
3. Alice and the server have a classical interaction over nm rounds. In each round,
   (a) Alice computes an angle δ_{ij} as a function of θ_{ij}, φ_{ij}, private randomness $r_{ij}\in\{0,1\}$, and previous outcomes b_{ij} reported by the server. She sends δ_{ij} to the server.
   (b) The server measures the (i,j)-th qubit of the brickwork state in the $\{|+\rangle_{\varphi_{ij}},|-\rangle_{\varphi_{ij}}\}$ basis and reports the outcome $b_{ij}\in\{0,1\}$ to the client.
4. Alice infers the outcome of her circuit from her private data and the server's last reported outcome.

There are many details missing to fully specify the protocol. The idea is to design rules for Alice to update the measurement angles δ_{ij} that she sends to the server in a way that, from the point of view of the server, δ_{ij} is alwyas uniformly random in D (so it reveals no information about the computation being performed), yet Alice is able to keep track of the actual computation being performed under her one-time pad. To see how this can be done, first consider the following exercise.

Exercise 10.3.2 Based on Exercise 10.3.1 we know that applying a Hadamard gate to a qubit A can be performed by measuring the qubit in the basis $\{|+\rangle,|-\rangle\}$, and adding an X^m correction, where m is the measurement outcome.

This is correct when the second qubit, B has been initialized in a $|+\rangle$ state, as it would be for the un-hidden brickwork state. Now suppose the qubit has in fact been initialized in the state $|+_\theta\rangle$, for some real angle θ (and a CTL-Z operation has been performed on the two qubits).

Show that the result of measuring the first qubit in the basis $\{|+_\delta\rangle, |-_\delta\rangle\}$ is to project the second qubit on $X^m H U_z(\theta+\delta)|\psi\rangle$, where m is the measurement outcome. ■

Suppose then that Alice would like to apply a gate $U_z(\varphi)$, for some angle $\varphi \in A$. The exercise shows that by communicating the angle $\delta = \varphi - \theta$ to the server instead, where θ is the initial angle using which she prepared the corresponding qubit of the brickwork state, will have the desired effect of implementing $X^m H U_z(\varphi)$. However, this still poses a problem: if the server is given both the quantum state $|+_\theta\rangle$, *and* the real angle $\varphi - \theta$, we can't argue that the computation is blind, as the joint distribution of these two pieces of information depends on φ.

Exercise 10.3.3 Fix φ, and suppose an adversary is given a classical value $\eta = \varphi - \theta$ and a single-qubit state $|\psi\rangle = |+_\theta\rangle$, where θ is chosen uniformly at random. Design a strategy for the adversary to recover φ, given $(\eta, |\psi\rangle)$. What is its success probability (averaged over the random choice of θ)? ■

The role of the additional values r_{ij} specified in the protocol is to render φ_{ij} completely hidden to the server. Here r_{ij} is chosen uniformly at random in $\{0,1\}$, and Alice communicates the angle $\varphi - \theta + r\pi$ to the server. Based on exercise 10.3.2 the effect of $r\pi$ on the computation is to add an extra Z^r correction, which Alice can easily keep track of. To see that it is sufficient to ensure blindness, imagine that instead $r\pi$ had been added to the initial angle θ. For any fixed θ, a random choice of $r \in \{0,1\}$ is sufficient to ensure the server gains no information from receiving $|+_{\theta+r\pi}\rangle$, as $\frac{1}{2}|+_\theta\rangle\langle+_\theta| + \frac{1}{2}|+_{\theta+\pi}\rangle\langle+_{\theta+\pi}| = \frac{1}{2}\mathbb{I}$. But as θ varies in D the angle $\theta - \varphi$ itself is uniformly distributed in D. Therefore from the point of view of the server the joint distribution of the pair $(|+_\theta + r\pi\rangle, \theta - \varphi)$ is indistinguishable from that of a uniformly random qubit and a uniformly random value from D. The server receives completely random data, so the computation is perfectly blind.

10.3.3 Verifiability

In the previous section we showed that blind delegation could be implemented in the MBQC model. Can we make the protocol verifiable? Note that so far the client does not perform any checks, so the server could just as easily report random outcomes to the client at each step. Already though, due to blindness there is no way the server can *force* a particular outcome on the client; the best it can do is mislead her into thinking that the outcome of the computation is some random bit.

There are different techniques available to make the protocol verifiable, and we refer to [FK12] for details. The main idea is to introduce *trap qubits*. Those are particular rows of the brickwork state that the client randomly inserts into its circuit but on which the only operation performed is a sequence of identity gates: they are meant to remain in the $|0\rangle$ state (hidden, as usual, under the quantum one-time pad). By asking the server to measure a qubit on such a line the client can verify the measurement outcome. Due to the blindness property, even the application of identity gates cannot be detected by the server, thus it does not know it is being tested.

Implementing this idea requires a little care, as it is important to ensure that even the tiniest attack by the server, such as reporting a single false measurement outcome, is detected with good probability: a single such deviation could suffice to ruin the whole computation. This can be achieved by introducing ideas from fault-tolerant computation that we will not go into here.

10.4 Delegating to two servers

Both schemes for delegated computation we've seen so far, in the circuit model or using measurement-based computation, require the client to prepare single-qubit states taken from a small fixed set and send them to the server. What if the client has no quantum abilities whatsoever? Intuitively, the aim

of the qubits sent by the client in the two previous schemes is to establish some kind of "trusted space" within the server's quantum memory, in which the computation is to be performed. The quantum one-time pad is used to guarantee that if the server tries to cheat by not using the qubits as required then the client interprets the results, at best, as garbage. In fact the verifiability property, enforced through the use of trap qubits, ensures that the server's cheating will be detected with high probability.

How can we establish a "trusted computation space" within the server's memory, without sending it the qubits in the first place? You know the answer! In previous weeks we've seen that simple tests based on the CHSH game could be used to guarantee that *two* arbitrary but non-communicating devices share a specific state, the EPR pair $|\phi^+\rangle$. Even if it is limited to a single qubit per server, this gives us a solid starting point: a test which ensures that a certain little corner of the servers' workspace behaves in a way that we can control.

Let's see how this idea can be leveraged to devise a scheme for delegated computation in which the verifier is completely classical, but has access to *two* non-communicating servers, both untrusted. This method is the most technical of the three we are presenting, and we'll remain at an intuitive level of presentation. If you are interested to learn more we refer you to the main paper on the topic [RUV13].

10.4.1 Establishing a trusted computation space

In Week 7 we saw the CHSH rigidity theorem, which states that if the servers successfully play the CHSH game then up to local isometries the operations they perform are equivalent to those specified in the ideal strategy for the CHSH game. Thus the CHSH game provides a simple test, not only to certify the presence of an EPR pair between the servers, but also the specific measurements that the servers perform on their respective half of the EPR pair when asked certain questions. The central idea for using this in delegated computation will be to alternate between playing the CHSH game with the servers, and playing other games, some of which involve the actual computation Alice wants the servers to implement; this will be done in a way that the servers individually can never tell whether they are being "CHSH-tested" or actually "used" to implement a useful part of the computation. Therefore the servers have to apply the honest CHSH strategy all the time, test or computation, and this gives us a way to control which operations they apply.

The first thing to deal with is that we're going to need many EPR pairs. One idea to certify n EPR pairs would be to play n CHSH games "in parallel": Alice could select n pairs of questions $(x_j, y_j)_{j=1,\ldots,n}$ to send to the servers, collect n pairs of answers (a_j, b_j), and check how many satisfy the CHSH condition $a_j \oplus b_j = x_j \cdot y_j$. If this estimate is close enough to the optimal $\cos^2(\pi/8) \cdot n$ she would accept the interaction. Although this is a sensible idea it is currently not known how well it works; in particular the effect of small errors in the servers' answers is not clear. (The difficulty is similar to one we encountered in Week 4, when we saw an example of a game for which the servers could play two repetitions of the game much better than you'd expect by using a correlated strategy across both instances of the game.)

Instead of executing the games in parallel Alice will perform them sequentially. That is, she sends the questions (x_j, y_j) to the servers one pair at a time, waiting for their answer before sending the next pair of questions. After having repeated this procedure for n rounds, she counts the number of rounds in which the CHSH condition was satisfied, and accepts if and only if it is at least $[\cos^2(\pi/8) - \delta]n$, for some error threshold δ. The following sequential rigidity theorem states the consequences of this test in the idealized setting where $\delta = 0$.

Theorem 10.4.1 — idealized. Suppose the two servers, Bob and Charlie, successfully play n sequential CHSH games. Then up to local isometries their initial state is equivalent to

$|\phi^+\rangle_{BC}^{\otimes n} \otimes |junk\rangle_{BC}$. Moreover, at each step $j \in \{1,\ldots,n\}$ the measurements performed by each server are equivalent to those of the ideal strategy for CHSH (Z and X for Bob and H and $\tilde{H}$ for Charlie) applied on the j-th EPR pair.

You may notice that the protocol for the n sequential CHSH tests is similar to how the CHSH tests are performed in the protocol for device-independent quantum key distribution we saw in Week 7. The analysis uses similar tools: a first step uses a (Martingale) concentration inequality to argue that, if a fraction about $\cos^2(\pi/8) - \delta$ of the games are won by the servers, then for most $j \in \{1,\ldots,n\}$ the *a priori* probability that the servers would have won in round j must be of the same order, say at least $\cos^2(\pi/8) - 2\delta$. For any such j the basic CHSH rigidity theorem can be applied to conclude that the measurements applied, and the state on which they were applied, are (up to local isometries) equivalent to the ideal CHSH strategy.

By itself this reasoning is not sufficient to imply that the servers' initial state is a tensor product of EPR pairs. Indeed, the different EPR pairs used in each round could partially "overlap", or even be the same pair! Intuitively we know this is not possible, as any measurement destroys the EPR pair, so it cannot be re-used. But this is tricky to establish rigorously; the way the errors add up through a proof by induction can be hard to control. Nevertheless, it can be done, and for the remainder of the section we will assume that a "robust" version of the "idealized" theorem above can be proven dealing with the more realistic setting where the servers are not required to play the CHSH games strictly optimally, a far too stringent requirement for any practical application.

10.4.2 State tomography

Now that we have a way to establish a "secure computation space", as a second step let's see how the client Alice can use that space, and additional CHSH tests, to certify that one of the servers has prepared certain single- or two-qubit states in that space.

Consider the following protocol. With Bob, Alice behaves exactly as if she was executing the n sequential CHSH games described in the previous section. With Charlie, however, she does something different: she instructs him to measure each half of the EPR pairs he is supposed to share with Bob in a certain basis, say $\{|+_\theta\rangle, |-_\theta\rangle\}$ for some real θ (defined as in (10.2)), and to report the outcome.

Charlie of course knows that something special is going on. So we have no guarantee as to what action he performs. In contrast, Bob is told the exact same thing as in the n-sequential CHSH test. He must thus behave exactly as if this is the test Alice was performing, and Theorem 10.4.1 applies: in each round, Bob applies the ideal CHSH measurements, in the standard or Hadamard bases, on his half of the j-th EPR pair, in a way that, if Charlie had been measuring using his own CHSH measurements, they would have succeeded with near-optimal probability.

But now Charlie is doing something different — we don't know what. But *if* Charlie performs the measurement asked by Alice, and reports the right outcome, we know what should happen: Charlie's half-EPR pair gets projected onto one of the basis states, $|+_\theta\rangle$ or $|-_\theta\rangle$, and by the special properties of EPR pairs so does Bob's half. In particular, whenever Bob performs a measurement in the Hadamard basis the average value of his outcome (considered as a value in $\{\pm 1\}$) should be $\langle +_\theta | X | +_\theta \rangle = \cos(\theta)$ or $\langle -_\theta | X | -_\theta \rangle = -\cos(\theta)$. Thus by collecting all Bob's answers associated to measurements in the X basis Alice can check whether the average outcome over the rounds in which Charlie reported a $+$ is approximately $\cos(\theta)$, and $-\cos(\theta)$ over those rounds when Charlie reported a $-$. Alice is using Bob's answers to perform tomography on the state that Charlie claims to have prepared, without Bob being able to detect what is going on! (Even though Bob knows he *might* be currently tested, since he is aware of the structure of the protocol, there is nothing he can do about it — if he deviates he risks failing too many CHSH games, in case this is what Alice is doing.)

In the CHSH game the only measurements made by Bob are in the computational or Hadamard

bases. To perform tomography of arbitrary multi-qubit states we would also need him to sometimes apply a Pauli Y. It is possible to do this via a simple modification of the CHSH game. For our purposes the modification will not be necessary, as the set of states that are characterized by their expectation value with respect to Pauli X and Z observables (we call such states *XZ-determined*) is sufficient to implement the delegated computation protocol.

Exercise 10.4.1 Show that the family of all single-qubit states in the xz-plane of the Bloch sphere, i.e. all states of the form

$$\rho = \frac{1}{2}[]I + \cos(\theta)X + \sin(\theta)Z], \qquad \theta \in [0, 2\pi),$$

are XZ-determined.

Show that the family of two-qubit states of the form

$$|\psi\rangle = U \otimes P|\phi^+\rangle,$$

for any single-qubit real unitary U and $P \in \{I, X, Y, Z\}$, is XZ-determined.

Give an example of two distinct single-qubit states that have the same expectation values with respect to both X and Z observables, and are thus not XZ-determined. ■

10.4.3 Process tomography

Beyond state tomography, our protocol for delegated computation will require us to implement some limited form of *process tomography*: we need to find a way to guarantee that at least one of the servers, Bob or Charlie, is performing the right computation! At first this task may appear overwhelming: while as described in the previous section it is possible to use one server to perform tomography against the other server's state, how can we test for a certain *gate* being applied? For the case of state preparation we know what the right states are, and as long as they are restricted to simple single- or two-qubit states we can do full state tomography. But our ultimate goal is to implement an arbitrary quantum circuit, which may generate highly entangled states of its n qubits; there is no hope to perform full tomography on such states, as it would require an exponential number of measurements.

We will sidestep the difficulty and use a model of computation which only requires the application of a very special type of gate — a measurement in the Bell basis, i.e. the simultaneous eigenbasis of $X \otimes X$ and $Z \otimes Z$, given by

$$|\psi_{00}\rangle_{AB} = \frac{1}{\sqrt{2}}(|00\rangle_{AB} + |11\rangle_{AB}), \qquad |\psi_{01}\rangle_{AB} = \frac{1}{\sqrt{2}}(|00\rangle_{AB} - |11\rangle_{AB}),$$
$$|\psi_{10}\rangle_{AB} = \frac{1}{\sqrt{2}}(|01\rangle_{AB} + |10\rangle_{AB}), \qquad |\psi_{11}\rangle_{AB} = \frac{1}{\sqrt{2}}(|01\rangle_{AB} - |10\rangle_{AB}),$$

where $|\psi_{00}\rangle = |\phi^+\rangle$ is the familiar EPR pair. This model of computation is called *teleportation-based computation* (recall that a measurement in the Bell basis is precisely the operation required of the sender in the teleportation protocol), and we'll review it in the next section. But let's already see how it can be used for delegated computation.

In a similar vein as in the previous section, suppose Alice instructs Charlie to measure his n qubits in the Bell basis, where the qubits are paired in an arbitrary way chosen by Alice (so she tells Charlie the whole set of measurements to be performed at the outset). Of course as usual Charlie does what he wants — he may not even have n qubits in the first place. But Alice also instructs Bob to play sequential CHSH games, so that from his point of view the protocol is perfectly indistinguishable from the tests. Once Alice has collected all of Bob and Charlie's outcomes, she

groups Bob's outcomes when they are associated to the same state, and uses them to check that Charlie did not lie. For instance, if Charlie reports $|\psi_{00}\rangle$ then whenever Bob measured the two corresponding qubits using the same basis, computational or Hadamard, his two outcomes should be the same. (Note that not all Bob's measurements are useful, as it will sometimes be the case that the qubits were measured in different bases, in which case there is no useful test Alice can perform — she simply discards those rounds.)

The following exercise asks you to make this argument more formal.

Exercise 10.4.2 Suppose Bob and Charlie share two EPR pairs, $|\phi^+\rangle_{B_1C_1} \otimes |\phi^+\rangle_{B_2C_2}$. Charlie measures his two halves, C_1C_2, using an arbitrary four-outcome POVM, obtaining a result $(c_1,c_2) \in \{0,1\}^2$. Bob measures each of B_1 and B_2 using observables $O_1, O_2 \in \{X,Z\}$ chosen uniformly at random.

Suppose that if $(O_1,O_2) = (X,X)$ then Bob's outcomes (as values in $\{\pm 1\}$) satisfy $b_1 b_2 = a$, and if $(O_1,O_2) = (Z,Z)$ they satisfy $b_1 b_2 = d$, for some fixed values $a,d \in \{\pm 1\}$ (i.e. imagine the same experiment is repeated many times, and Bob's outcomes consistently satisfy these equations, for the same values of a and d). Show that Charlie must have been implementing a measurement in the Bell basis. Which Bell state is associated to each of the four possible values for (a,d)? ∎

The exercise shows that, provided we can trust that Bob and Charlie indeed share EPR pairs, and Bob's measurements are made in the computational or Hadamard bases, then Alice has a way to verify that Charlie has been implementing a Bell basis measurement on certain pre-specified pairs of qubits. Just as for the case of state tomography, these assumptions are guaranteed by the fact that Bob cannot tell the difference between when Alice is executing the process tomography protocol described here, or when she is executing sequential CHSH games.

10.4.4 Teleportation-based computation

The final ingredient needed for our delegation protocol is a method of computation adapted to the kinds of operations we are able to certify of the servers: preparation of EPR pairs and single- or two-qubit XZ-determined states (Exercise 10.4.1), and measurements of pairs of qubits in the Bell basis (Exercise 10.4.2).

Computation by teleportation is a model of computation which allows just that. The main idea is that a gate can be applied to a qubit by "teleporting the qubit into the gate". The following exercise fleshes out the main gadget used in computation by teleportation.

Exercise 10.4.3 Let $|\psi\rangle_A$ be an arbitrary single-qubit state and let $|\phi\rangle_{BC} = (I \otimes UP|\phi^+\rangle)$, where U is an arbitrary single-qubit unitary and $P \in \{I,X,Y,Z\}$. Suppose a measurement of qubits A and B is performed in the Bell basis, yielding a pair of outcomes $(b_1,b_2) \in \{0,1\}^2$. Show that there exists a Pauli operator Q (depending only on (b_1,b_2)) such that the post-measurement state of the qubit in C is $(UQPU^\dagger)U|\psi\rangle$. ∎

The idea is then the following. Suppose that Alice wishes to implement an arbitrary computation on n qubits, specified by a circuit $\mathscr{C}$ using the universal gate set $\mathscr{G} = \{\text{CNOT}, G\}$ introduced in Section 10.1.1. Assume for simplicity the input to the circuit is $|0\rangle^{\otimes n}$. Alice initializes her work space with a large number of "magic states" from the set

$$\big\{|0\rangle,\ (I\otimes H)|\phi^+\rangle,\ (I\otimes G)|\phi^+\rangle,\ \text{CNOT}_{B_1B_2}(|\phi^+\rangle_{A_1B_1}|\phi^+\rangle_{A_2B_2})\big\}. \tag{10.4}$$

At each stage of the computation Alice keeps track of a special set of n qubits which represent the current state of the circuit. We can label these as $A_1 \cdots A_n$, even though they will change over time. Initially $A_1 \cdots A_n$ point to any n of the "magic" $|0\rangle$ qubits she has prepared in her workspace.

Now suppose Alice would like to apply a gate to one of her qubits A_j, for example a G gate. Then she can perform the circuit described in Exercise 10.4.3, where the role of A is played by A_j, and the roles of B and C by one of her "magic" $(I\otimes G)|\phi^+\rangle$. As a result the state of C is projected to $(GQG^\dagger)G|\psi\rangle_C$, where initially A_j is in state $|\psi\rangle$ (the same computation would work for mixed states as well). This is the operation Alice wanted to perform, except for the correction $GQG^\dagger$. How do we deal with this?

Depending on Q, $GQG^\dagger$ will amount to a Pauli correction, possibly multiplied by a Hadamard: $GXG^\dagger = iHY$, $GYG^\dagger = Y$ and $GZG^\dagger = H$. By now we are used to Pauli corrections: Alice can keep track of these as a form of one-time pad that is tagged along the whole computation. The Hadamard gate is a little more annoying, but in fact it can be easily corrected using one more step of "teleportation", this time using a "magic" $(I\otimes H)|\phi^+\rangle$. This will induce yet another correction $HQ'H^\dagger$, but this time whatever Q' is the result is a Pauli correction that Alice can again tag along as part of the one-time pad.

Thus, aside from the preparation of the magic states, the whole computation boils down to a simple sequence of Bell basis measurements. Note however that, due to the necessity of performing Hadamard corrections in an unpredictable way (as it depends on measurement outcomes obtained when teleporting into a G gate), this sequence is adaptive. This is similar to the scenario of MBQC, but it will require us to proceed with a little extra care in the final delegation protocol.

10.4.5 Blind and verifiable delegated computation

With all the ingredients in place we are ready to describe a delegation protocol for the classical client Alice to delegate the computation of an n-qubit quantum circuit $\mathscr{C}$, specified using the universal gate set $\{\text{CNOT}, G\}$, to two isolated quantum servers, Bob and Charlie, sharing entanglement.

Protocol 2 Alice compiles a description of $\mathscr{C}$ in the teleportation-based model. Let $t = \text{poly}(|\mathscr{C}|)$ be chosen large enough, and $\delta = \text{poly}^{-1}(|\mathscr{C}|)$ small enough. Alice performs one of the following four tests (the first three with equal probability, and the last with smaller probability depending on the size of $\mathscr{C}$):

1. She plays t sequential CHSH games with the two servers. If the fraction of successes is below $\cos^2(\pi/8) - \delta$ she rejects.
2. Alice executes a state tomography protocol in which Charlie is asked to perform measurements that correspond to $t/9$ copies of each of the four magic states in (10.4). Bob is instructed to play t CHSH games. Bob's measurement outcomes are used to check the results reported by Charlie, as described in Section 10.4.2.
3. Alice executes a process tomography protocol in which Bob is instructed to perform Bell basis measurements according to the pattern that corresponds to the teleportation-based circuit associated with $\mathscr{C}$. Charlie is instructed to play t CHSH games. Charlie's measurement outcomes are used to verify that Bob is reporting the correct outcomes for his Bell measurements, as described in Section 10.4.3.
4. Alice orchestrates the computation of $\mathscr{C}$: she instructs Charlie to prepare the magic states as in step 2., and Bob to perform Bell basis measurements as in step 3. She uses the measurement outcome reported by Bob for the output qubit as the output of the computation (after having applied any required Pauli corrections).

The main reason this protocol works is that Bob cannot distinguish a computation run from a process tomography run, or a CHSH run from a state tomography run; similarly Charlie cannot distinguish between a computation run and a state tomography run, or between a CHSH run and a process tomography run. The protocol can be made blind: blind to Bob, whom without knowledge of which magic states his Bell basis measurements are performed on gains no useful information from the pattern of Bell measurements Alice instructs him to perform (the pattern can be made

independent from the circuit $\mathscr{C}$, aside from its size); blind to Charlie, who prepares magic states in a way that is completely independent from the computation. Verifiability follows directly from the tests performed in cases 1., 2. and 3. of the protocol.

There is one difficulty we hinted at earlier and we have glossed over so far. This is the fact that, after application of a Bell basis measurement corresponding to teleportation into a G gate, Alice needs to make an adaptive choice: either apply an H correction, or not. However, Bob should be ignorant of this choice, as otherwise the protocol would no longer be blind. The solution is to switch the focus over to Charlie. While Bob will always be asked to perform the same pattern of Bell basis measurements, Charlie will be (adaptively) asked by Alice to create certain magic states as $|\phi^+\rangle$, and others as $(I \otimes H)|\phi^+\rangle$, as a function of the outcomes reported by Bob. Since these outcomes are uniformly distributed, the pattern of state preparation requests Charlie sees is still random, so that he does not gain any information about the computation either. (Note however that a third observer able to eavesdrop on both the messages exchanged with Bob and with Charlie would learn valuable information about the computation; however such an attack falls outside of the scope of the security definition of delegated computation.)

Only one task remains: performing a soundness analysis of the protocol! Given that it is not possible to require that the servers *exactly* pass all the tests, some error should be tolerated. How does this error effect the quality and trustworthiness of the computation? This is quite delicate. The best analysis known to-date makes this protocol, compared to the ones we saw in the previous two sections, highly inefficient, as it requires T to be a very large power of n before even relatively weak security guarantees can be obtained. Nevertheless, it is the only protocol known for purely classical delegated computation, and improving it is an important research problem.

Acknowledgments

 The lecture notes are written by Nelly Ng, Thomas Vidick and Stephanie Wehner. We thank David Elkouss, Kenneth Goodenough, Jonas Helsen, Jérémy Ribeiro, and Andrea Coladangelo for proofreading.

Bibliography

[Aha+08] Dorit Aharonov et al. “Adiabatic quantum computation is equivalent to standard quantum computation”. In: *SIAM review* 50.4 (2008), pages 755–787 (cited on page 10).

[BB06] Dan E Browne and Hans J Briegel. “One-way quantum computation-a tutorial introduction”. In: *arXiv preprint quant-ph/0603226* (2006) (cited on page 12).

[Bro15] Anne Broadbent. “How to verify a quantum computation”. In: *arXiv preprint arXiv:1509.09180* (2015) (cited on pages 7, 8).

[Dun+14] Vedran Dunjko et al. “Composable security of delegated quantum computation”. In: *International Conference on the Theory and Application of Cryptology and Information Security*. Springer. 2014, pages 406–425 (cited on page 4).

[Fey82] Richard P Feynman. “Simulating physics with computers”. In: *International journal of theoretical physics* 21.6 (1982), pages 467–488 (cited on page 3).

[FK12] Joseph F Fitzsimons and Elham Kashefi. “Unconditionally verifiable blind computation”. In: *arXiv preprint arXiv:1203.5217* (2012) (cited on page 13).

[RUV13] Ben W Reichardt, Falk Unger, and Umesh Vazirani. “A classical leash for a quantum system: command of quantum systems via rigidity of CHSH games”. In: *Proceedings of the 4th conference on Innovations in Theoretical Computer Science*. ACM. 2013, pages 321–322 (cited on page 14).