Quantum Cryptography and Security

Prof. Pino Vallone, Prof. Nicola Laurenti

a.a. 2021/2022

A. Marcomini
S. Piccinelli
T. Faorlin

Contents

Compiled on: 20th January, 2022.

Chapter 1

Recap of Quantum Mechanics

Quantum Mechanics (QM) is one of the major breakthrough in Physics of the last century. Like in classical mechanics, the laws in this field have been derived with the effort of many people, but when a student attends General Physics 1 at University, the professor is not teaching him things following a temporal line, or highlighting how different ideas coming from different brilliant minds have in the end generated a beautiful theory. The professor will simply write $\vec{F} = m\vec{a}$ and that's it. In this course, we are going to do the same with Quantum Mechanics.

This chapter is intended as a rough summary of some basic concepts, needed to lay the foundations for future lectures. For more information we refer to other material.

1 Pure states formalism

States

Quantum states, or wave-functions, live in a linear vector space called Hilbert space $\mathcal{H}$, and we make use of Dirac's notation to identify them. In particular, $|\psi\rangle$ is a **ket** and $\langle\psi| = |\psi\rangle^\dagger$ is a **bra**. Given a $d-$dimensional space we can define a basis $\{|0\rangle, \dots, |d-1\rangle\}$ and rewrite the quantum state in terms of the basis vectors

$$|\psi\rangle = \sum_{n=0}^{d-1} \psi_n |n\rangle \qquad\qquad \langle\phi| = \sum_{n=0}^{d-1} \phi_n^\star \langle n| \tag{1.1}$$

In $d = 2$ this is exactly a two-level quantum system, also known as **qubit** $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$. In the latter is contained all the difference from a classical bit: the quantum state $|\psi\rangle$ describing the qubit is a **quantum superposition** of two basis vectors $|0\rangle$ and $|1\rangle$.

In Eq. (1.1), the $\psi_n = \langle n|\psi\rangle \in \mathbb{C}$ coefficients are **complex amplitudes**, and if we take their modulus squared, following Born's rule, we obtain the **probability** to

find the system described by $|\psi\rangle$ in the state $|n\rangle$. A requirement on a quantum state is that it must be **normalised**, meaning that

$$\sum_{n=0}^{d-1} |\psi_n|^2 = 1 \tag{1.2}$$

Furthermore, in this space is also defined a **scalar product** between two states

$$\langle\phi|\psi\rangle = \sum_{n=0}^{d-1} \phi_n^\star \psi_n \langle n|n\rangle = \sum_{n=0}^{d-1} \phi_n^\star \psi_n \tag{1.3}$$

Evolution

We now look at how the state changes in time. The evolution for pure states is described in terms of **unitary operators** $\hat{U}$, so that $|\psi\rangle \longrightarrow \hat{U}|\psi\rangle \equiv |\psi'\rangle$. The unitary property states that $\hat{U}^\dagger \hat{U} = \mathbb{1}$ and this preserves the scalar product

$$\langle\phi'|\psi'\rangle = \langle\phi| U^\dagger U |\psi\rangle = \langle\phi|\psi\rangle$$

Measure

In classical physics we measure variables, and in quantum mechanics we do the same by means of hermitian operators ($\hat{O}^\dagger = \hat{O}$). These operators can be thought as matrices and are also called **observables**. To understand better how they work we need to use their spectral decomposition. In fact, any hermitian operator can be put in diagonal form

$$\hat{O} = \sum_{n=0}^{d-1} \lambda_n |\lambda_n\rangle \langle\lambda_n| = \sum_{n=0}^{d-1} \lambda_n \hat{\Pi}_n \tag{1.4}$$

where λ_n and $|\lambda_n\rangle$ are respectively eigenvalues and eigenvectors of the matrix $\hat{O}$ and $\langle\lambda_i|\lambda_j\rangle = \delta_{ij}$. Thanks again to hermiticity we have that $\lambda_n \in \mathbb{R}$. The eigenvalues are the possible outcomes of a measurement. Born's rule allows us to determine which is the probability that, measuring a quantum state we end up measuring exactly λ_n, namely

$$P_{\lambda_n} = |\langle\lambda_n|\psi\rangle|^2 = \langle\lambda_n|\psi\rangle \langle\psi|\lambda_n\rangle = \mathrm{Tr}\left[|\psi\rangle\langle\psi| \underbrace{|\lambda_n\rangle\langle\lambda_n|}_{\hat{\Pi}_n}\right] \tag{1.5}$$

Indeed, from the definition of trace, given a generic basis $\{\lambda_j\}_{j=0}^{d-1}$ of $\mathcal{H}$ (since the trace is invariant under this choice),

$$\begin{aligned}\mathrm{Tr}\left(|\psi\rangle\langle\psi|\lambda_n\rangle\langle\lambda_n|\right) &= \sum_{j=0}^{d-1}\langle\lambda_j|\psi\rangle\langle\psi|\lambda_n\rangle\langle\lambda_n|\lambda_j\rangle = \\ &= \sum_{j=0}^{d-1}\langle\lambda_j|\psi\rangle\langle\psi|\lambda_n\rangle\,\delta_{jn} = \langle\lambda_n|\psi\rangle\langle\psi|\lambda_n\rangle\end{aligned}$$

Alternatively, we can use the cyclic property of the trace, namely $\mathrm{Tr}\left(\hat{A}\hat{B}\hat{C}\right) = \mathrm{Tr}\left(\hat{B}\hat{C}\hat{A}\right)$. In Eq. (1.5), $\hat{\Pi}_n$ is the projector ($\hat{\Pi}_n^2 = \hat{\Pi}_n$) in the eigenspace correspondent to the eigenvalue λ_n ($\hat{\Pi}_n \equiv |\lambda_n\rangle\langle\lambda_n|$) and will help us in the last point of the recap on pure states: the collapse.

Collapse

In QM, after we have measured a state, the latter collapses to its projection, thanks to the Von Neumann theorem, along the eigenvector that corresponds to the eigenvalue resulting from the measurement

$$|\lambda_n\rangle = \frac{\hat{\Pi}_n|\psi\rangle}{\sqrt{P_{\lambda_n}}} = \frac{\langle\lambda_n|\psi\rangle}{\sqrt{P_{\lambda_n}}}|\lambda_n\rangle = e^{i\phi}|\lambda_n\rangle \tag{1.6}$$

We can see the projection operation as the adding of a global phase in front of the state. Specifically, in the vector space $|\psi\rangle \sim e^{i\phi}|\psi\rangle$ are two different states **but** in a QM framework they represent the same state.

2 Mixed states formalism

States

When dealing with real problems, it is more common to deal with non-isolated systems rather than perfectly isolated ones. In these situations, we must replace the state vector $|\psi\rangle$ with a **density matrix** $\hat{\rho}$ (an operator). In this case we do not know the state of our system, but we know that it can come from an ensemble of possibilities $\{|\psi\rangle_i\}_{i=1\ldots N}$ each with a defined probability p_i ($p_i > 0$, $\sum p_i = 1$). As a remark, the states in the ensemble have nothing to do with the Hilbert space ($N > d \equiv \dim(\mathcal{H})$ in principle) and they are not orthogonal. Formally,

$$\hat{\rho} = \sum_{n=1}^{N} p_n |\psi_n\rangle\langle\psi_n|\,; \tag{1.7}$$

if $\hat{\rho}$ is describing a pure state then the whole definition reduces to a simple projector on the state $\rho = |\psi\rangle\langle\psi|$. In this case, the phase invariance presented in the last section is straightforward

$$\hat{\rho}' = |\psi'\rangle\langle\psi'| = e^{i\phi}|\psi\rangle\langle\psi|e^{-i\phi} = |\psi\rangle\langle\psi| = \hat{\rho}$$

The density matrix can be defined by its matrix elements in the following way: recalling Eq. (1.1),

$$\hat{\rho}_{ij} = \langle i|\hat{\rho}|j\rangle = \sum_{n=1}^{N} p_n \langle i|\psi_n\rangle\langle\psi_n|j\rangle = \sum_{n=1}^{N} p_n \psi_i^n (\psi_j^n)^\star$$

and it has three fundamental properties:

— $\hat{\boldsymbol{\rho}}$ **is hermitian**. This can be proven looking at the matrix elements and proving that $\hat{\rho}_{ij} = (\hat{\rho}_{ji})^\star$. In particular we have that

$$\hat{\rho}_{ij} = \sum_{n=1}^{N} p_n \psi_i^n (\psi_j^n)^\star = \sum_{n=1}^{N} p_n \psi_j^n (\psi_i^n)^\star = \sum_{n=1}^{N} p_n (\psi_j^n)^\star \psi_i^n = (\hat{\rho}_{ji})^\star$$

— $\mathrm{Tr}(\hat{\rho}) = \mathbf{1}$. This comes directly from the definition of the matrix elements

$$\mathrm{Tr}(\hat{\rho}) = \sum_{i=1}^{d} \langle i|\hat{\rho}|i\rangle = \sum_{n=1}^{N} p_n \sum_{i=1}^{d} |\psi_i^n|^2 =$$
$$= \sum_{n=1}^{N} p_n \sum_{i=1}^{d} \langle\psi_n|i\rangle\langle i|\psi_n\rangle = \sum_{n=1}^{N} p_n = 1$$

— $\hat{\boldsymbol{\rho}}$ **is a non-negative operator**. $\forall\,|\psi\rangle$ it holds $\langle\psi|\hat{\rho}|\psi\rangle \geq 0$.

Evolution

To describe the evolution of a density matrix, we need to put into the game some more general operators, called **superoperators** (since they are a transformation between operators) $\mathcal{E}$ that are acting on a $\hat{\rho}$ as

$$\hat{\rho} \longrightarrow \hat{\rho}' \equiv \mathcal{E}(\hat{\rho})$$

This kind of superoperators have different properties:

— $\boldsymbol{\mathcal{E}}$ **is linear**, $\mathcal{E}(a\hat{\rho}_a + b\hat{\rho}_b) = a\mathcal{E}(\hat{\rho}_a) + b\mathcal{E}(\hat{\rho}_b)$.

— $\mathrm{Tr}[\mathcal{E}(\hat{\rho})] = \mathrm{Tr}[\hat{\rho}]$, i.e. they are trace preserving.

— $\boldsymbol{\mathcal{E}(\hat{\rho})^\dagger = \mathcal{E}(\hat{\rho}^\dagger)}$, i.e. they preserve hermiticity. In other words, $\mathcal{E}(\hat{\rho})$ is hermitian if $\hat{\rho}$ is hermitian too.

— The evolution is **completely postive**. This includes simple positivity of course (if $\hat{\rho} \geq 0$ then $\mathcal{E}(\hat{\rho}) \geq 0$), but completely means that it is always possible to extend the the action of the superoperator to an action that acts locally on a system and leaves all the rest unchanged. In other words, the extension of $\mathcal{E}$ from $\mathcal{H}_1$ to $\mathcal{H}_1 \otimes \mathcal{H}_{env}$ is a positive map. This is of experimental interest: we will never deal with states of the entire universe! An example of an operation that is not completely positive is the **transposition**, which is not a physical operation. If a transposition operation is applied on a larger system the resulting state is not always positive.

Definition 2.1 (Kraus representation)**.** It can be shown that every superoperator $\mathcal{E}$ with the properties listed above can be written in Kraus representation, i.e. as "decomposed" sum of M terms,

$$\mathcal{E} : \hat{\rho} \mapsto \hat{\rho'} \equiv \mathcal{E}(\hat{\rho}) = \sum_{n=1}^{M} \hat{K}_n \rho \hat{K}_n^\dagger \qquad\qquad \sum_{n=1}^{M} \hat{K}_n^\dagger \hat{K}_n = \mathbb{1} \tag{1.8}$$

where $\{\hat{K}_n\}_{n=1\ldots M}$ are called **Kraus operators**. The converse implication is also valid. For a unitary evolution we will have in the end only one Kraus operator $\hat{K}$ (and look how the unitary condition is back). In general, is not easy to go from $\mathcal{E}(\hat{\rho})$ to $\hat{K}_n$, even because the Kraus decomposition is not unique. It can be demonstrated (**Neumark's Theorem**) that is always true that any generalized evolution (described by any classes of Kraus operators) can be seen as induced by a unitary evolution on a bigger system.

Measure

If we have a system composed of two subsystems 1 and 2 (**ancilla**), we can either perform a projective measurement on system 1 (using Von Neumann's formalism) or on 2. In the latter case this action induces a generalized measurement on system 1 due to the correlations produced in general by the unitary evolution. This opens the doors to new classes of measurements.
In summary, the idea is the following: given a qubit, we correlate it with an ancillary one and we perform a projective measurement on this second qubit. This action will automatically translate to a generalized measurement on 1: it is important to stress that in this way we are perturbing as less as possible the system under examination.

This process is called **weak measurement**, and can be seen in the framework of **generalized measurements**, a class of measurement processes with an uncertain

outcome, whereby the final state is an element $\{|\psi\rangle_i\}_{i=1,\dots,N}$ chosen with probability p_i. In place of the projector we then use the (larger) set of operators $\{\hat{M}_i\}$, not necessarily self-adjoint, which map $|\psi\rangle$ into the various possibilities $|\psi_i\rangle$.
Since we want for the $|\psi_i\rangle$ to span all the space of possible evolutions of $|\psi\rangle$, we require

$$\sum_{n=1}^{N} p_n = 1 \Leftrightarrow \sum_{n=1}^{N} \hat{M}_n^\dagger \hat{M}_n = \mathbb{1} \tag{1.9}$$

Collapse

Similar to what happens with pure state, also with density matrix $\hat{\rho}$ we can look for the **PSM** (**p**ost **m**easurement **s**tate). In particular, if the m-th output is obtained:

$$\hat{\rho} \longrightarrow \frac{\hat{M}_m \hat{\rho} \hat{M}_m^\dagger}{p_m} \equiv \hat{\rho}' \qquad\qquad p_m = \mathrm{Tr}\Big[\hat{M}_m^\dagger \hat{M}_m \rho\Big] \tag{1.10}$$

Remarkably, this is all QM can tell us.
If M_n are self-adjoint and $M_i M_j = \delta_{ij} M_i \Rightarrow M_i^2 = M_i$ we retrieve the special case of a pure state ($\hat{M}_n \equiv \hat{\Pi}_n$, $\sum_{n=1}^{N} \hat{M}_n = \mathbb{1}$). Again, pure state are contained in a more general treatment as a special case.

Finally, if we don't care about the final state our system will end up in but rather on the probability outcome of the measurement we can avoid defining the full measurement operators $\hat{M}_n$ (since they are needed only in the collapse); this can be formally treated by introducing a more generalized class of measures, i.e. **POVM** Measurement (**p**ositive **o**perator **v**alued **m**easurement). This is done by means of a set of non-negative operators $\{\hat{F}_n\}_{n=1\dots M}$, $\hat{F}_n \geq 0$, with the property

$$\sum_{n=1}^{M} \hat{F}_n = \mathbb{1} \tag{1.11}$$

Each F_i describes a possible outcome of a measure with associated probability given by its expectation value $p_n = \mathrm{Tr}\Big[\hat{F}_n \hat{\rho}\Big]$.
If the decomposition $\hat{F}_n = \hat{M}_n^\dagger \hat{M}_n$ holds, than we retrieve the case of the generalized measures (which in turn contain all possible projective measurements).
POVM are the most general class of measurements one can perform on a state.

As a final remark for this chapter, we recall that a qubit density matrix can be rewritten as $\hat{\rho} = \frac{1}{2}(\mathbb{1} + \vec{\sigma} \cdot \vec{r})$, where $r_i \in \mathbb{R}\ \forall i$ and $\|\vec{r}\| \leq 1$. If the last inequality is saturated, it means that we are dealing with pure states.

Chapter 2

Recap of Information Security

1 Terminology

Security goals

The desirable features of communications and networks are:

— **Confidentiality:** Information is available to the intended receiver only;

— **Integrity:** Information is received exactly as sent;

— **Availability:** Service is always available even if someone intends to disrupt the network;

— **Accountability:** It is always possible to identify who is responsible for any information event;

— **Privacy:** information is (used but) not disclosed to anyone (see conflict with accountability).

General attacks and threats

The main attacks that can threaten security goals are:

— **Eavesdropping:** Learning confidential information;

— **Modification:** Modifying a message in transit;

— **Denial of service:** Make the service unavailable (e.g., by keeping it busy);

— **Forgery:** Build a fake message, pretending it was sent by someone else;

— **Masquerade:** Posing as someone else in a single message transmission or interactive protocol;

— **Repudiation:** Denying having sent or received some message;

— **Profiling:** Gathering information about a single user;

— **Fingerprinting:** Identifying the user associated with some message;

— **Traffic analysis:** Learning origin, destination, length, times, of communications (not the content).

Security services

Security services should protect communications and network protocols from attacks. The principal ones are:

— **Secrecy:** Makes message unintelligible for eavesdroppers;

— **Integrity protection:** Makes it possible to detect whether a message was intercepted and modified;

— **Access control:** Can protect from denial of service;

— **Message authentication:** Allows to detect forged messages;

— **Entity authentication:** Protects from masquerades;

— **Non-repudiation:** Prevents repudiation by source and/or destination;

— **Anonymization:** Prevents consistent association between message and user;

— **Key management:** Ancillary services to cryptographic mechanisms.

Security mechanisms

A security mechanism is a way to implement some security service. Among them we find:

— **Encryption:** The message is transformed for communication and only the intended receiver can reverse the transform;

— **Digital signature:** A string is appended to the message that can only be created by the legitimate source but can be publicly verified;

— **Intrusion detection:** Identify characteristics of malicious behavior (signature) or non-typical legitimate behavior (anomaly);

— **Message authentication codes:** A string is appended to the message that can only be created by the legitimate transmitter or receiver;

— **Randomization:** Events of the same category are randomly permuted to break dependencies;

— **Key agreement:** Several agents cooperate to produce a key from inputs provided by each of them.

It is not difficult to observe that the definitions above are indeed interconnected. A summary of all these elements and their connections is reported in Fig. 2.1.

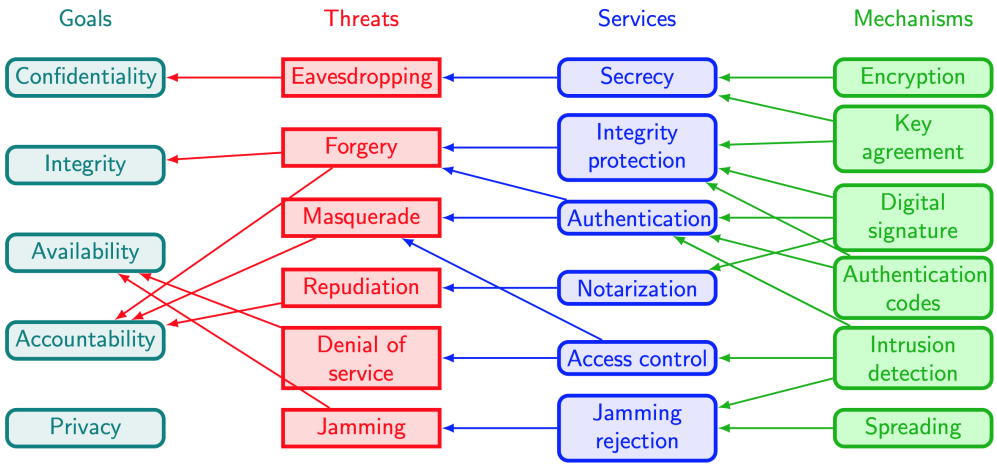


Figure 2.1: A summary of the main elements of information security and their interconnections.

2 Quantitative security definitions

Here we briefly try to formalize from a mathematical point of view the concept of information security and to derive some useful results.

2.1 Concrete formulation

Definition 2.1 (Single attack)**.** A single attack is modelled as a **randomized algorithm** A, characterized by a **success event** S_A and its **execution time** T_A (which is, in general, a random variable).

Definition 2.2 (Security mechanism)**.** A security mechanism is modelled as a **randomized algorithm** M, characterized by its **execution time** T_M (which may also be a random variable).

Definition 2.3 (Security measure vs single attack)**.** The security provided by mechanism M against attack A can be measured by the conditional probability

$$P\left[S_A; A, M\right]$$

that the success event S_A is achieved by attack A while implementing mechanism M.

Definition 2.3 allows to quantify in a quite intuitive way the security of a mechanism against a single attack. While aiming for a more general proposition, we can extend naturally this result.

Definition 2.4 (Security measure vs class of attacks)**.** Consider a class $\mathcal{A}$ of attacks with a common success event $S_{\mathcal{A}}$. Then, the security of a mechanism M against the class $\mathcal{A}$ of attacks is measured by

$$\sup_{A \in \mathcal{A}} P\left[S_{\mathcal{A}}; A, M\right]$$

It is sometimes convenient to take the binary estimation of the quantity above, which takes the name of **security level** of M against $\mathcal{A}$

$$SL(M) = \log_{1/2} \sup_{A \in \mathcal{A}} P\left[S_{\mathcal{A}}; A, M\right] \quad \text{[bits]}$$

Of course, we would desire a mechanism to be secure against the widest possible class of attacks.

Definition 2.5 (Unconditional security)**.** A security mechanism M is said to offer ε-unconditional security against a class $\mathcal{A}$ of attacks, for some $\varepsilon > 0$, if

$$P\left[S_{\mathcal{A}}; A, M\right] \leq \varepsilon, \ \forall \ A \in \mathcal{A}$$

That is, all attacks in $\mathcal{A}$ succeed against M with probability **no more than ε**.

Definition 2.6 (Computational security)**.** A security mechanism M is said to offer (ε, T_0)-computational security against a class $\mathcal{A}$ of attacks, for some $\varepsilon > 0$ and $T_0 > 0$, if

$$P\left[S_{\mathcal{A}} \cap \{T_A < T_0\}; A, M\right] \leq \varepsilon, \ \forall \ A \in \mathcal{A}$$

That is, all attacks in $\mathcal{A}$ succeed against M in a time shorter than T_0 with probability no more than ε. A visual representation of this is reported in Fig. 2.2: as we can see, this requirement defines a forbidden region for the time window $[0, T_0]$ that must not be crossed by the success probability of attacks in $\mathcal{A}$.

2.2 Asymptotic formulation

The definitions reported in the previous paragraph are indeed subject to technological maturity. To overcome this problem, it is common practice to allow the mechanism to depend on a security parameter $n \in \mathbb{N}$ (e.g., the length of cryptographic keys, the entropy of signatures, the number of rounds in an interactive protocol, etc.) that can be increased at will so that:

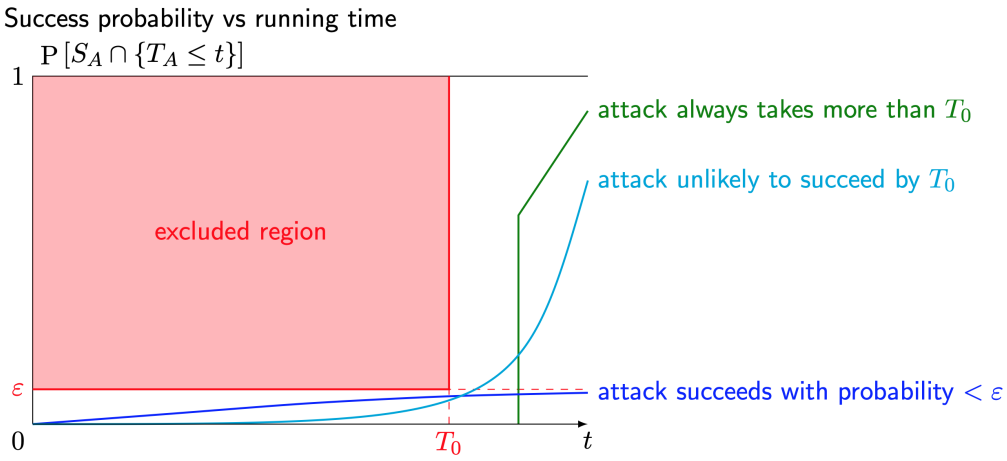


Figure 2.2: Visual representation of computational security requirement. The most of the cyan curve is outside the range of T_0, and this translates to an attack that is very unlikely to succeed by this amount of time.

— the legitimate operation is still feasible (complexity depends on n polynomially)

— the adversary operation soon becomes infeasible (superpolynomial complexity increase or success probability decrease)

Definition 2.7 (Asymptotic security)**.** A sequence of security mechanisms $\{M_n\}_n$, indexed by some parameter $n \in \mathbb{N}$, is said to offer asymptotic security against a class $\mathcal{A}$ of attacks, if:

1. $\exists\ p(\cdot)$ polynomial, such that

$$T_{M_n} \leq p(n)\,,\ \forall\ n$$

2. $\forall\ q(\cdot), s(\cdot)$ polynomials and sequence of attacks $\{A_n\}_n \subset \mathcal{A}$, $\exists\ n_0$ such that

$$P\left[S_{\mathcal{A}} \cap \{T_{A_n} < q(n)\}; A_n, M_n\right] \leq \frac{1}{s(n)},\ \forall\ n > n_0$$

It is also said that the probability of the attack succeeding in polynomial time vanishes super polynomially, i.e. that it is asymptotically negligible.

Fig. 2.3 depicts a visual interpretation of this behaviour: in a log-log scale, the polynomial complexity of $\{M_n\}$ is represented by a straight line defining the execution time growth with n (blue line on the left). On the right, we can see that either the execution time of $\{A_n\}_n$ scales polynomially but has superpolynomially vanishing

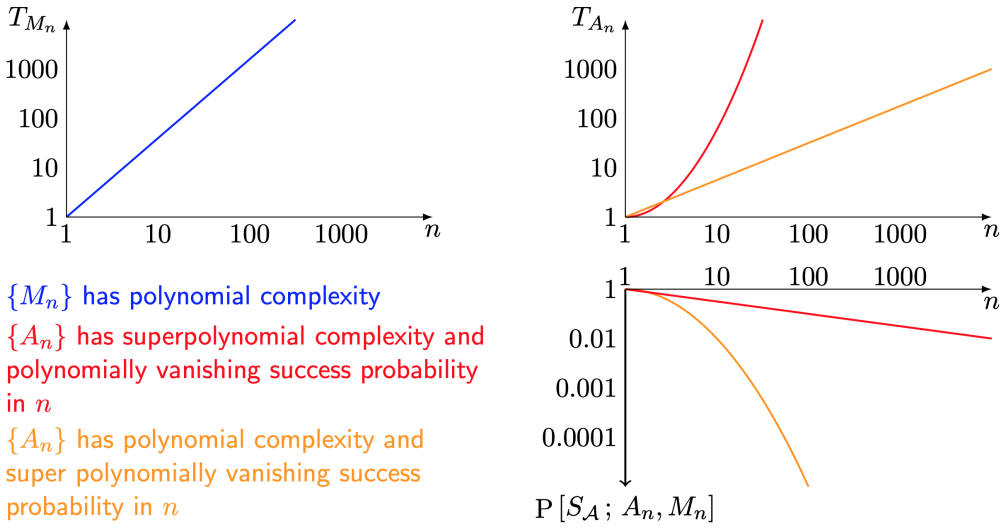


Figure 2.3: Graphical interpretation of asymptotic computational security.

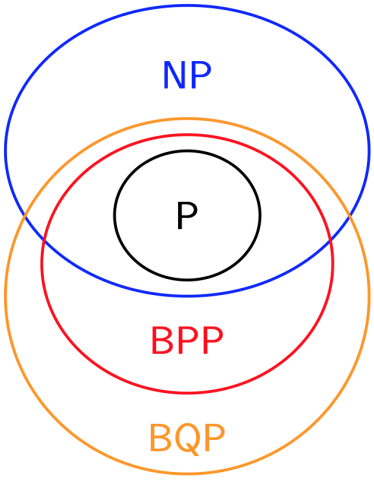
success probability (orange lines) or it has polynomial vanishing success probability at the price of allowing superpolynomial execution time growth. We can also formalize the definitions above in terms of complexity classes for decision problems. To give a brief recap, those are:

— P: decision problems that can be exactly solved by a deterministic algorithm in polynomial time;

— NP: decision problems that can be solved in the positive case by a deterministic algorithm in polynomial time with a polyonmial size certificate;

— BPP: decision problems that can be solved with "low" error probability by a probabilistic algorithm in polynomial time;

— BQP: decision problems that can be solved with "low" error probability by a quantum algorithm in polynomial time.



Then, recollecting what said above regarding the computational scaling of the quantities of interest, we can conclude that previous definitions correspond to:

$$\text{Asymptotic computational security} \iff \text{attacking } \{M_n\} \notin \text{BPP}$$

$$\text{Idem vs a quantum adversary} \iff \text{attacking } \{M_n\} \notin \text{BQP}$$

2.3 Composition of security mechanisms and distinguishability

Practically speaking, we usually deal with theoretical formulations of mechanisms (assumed perfect), while in reality we have to take into account **potential failure** probabilities. This to be said, at the end of the day our final scope is the estimation of the security of the **global implementation**, which could be achieved by **combining** several mechanisms, and that can be hard to be calculated as a whole. This leads to a question: can we estimate the security of a composed mechanism by means of partial security estimations of its components?
Mathematically speaking, we consider a security mechanism S which makes use of another mechanism M, denoting this occurrence $S[M]$. We now take into account the ideal counterpart of M, namely $M^\star$, to introduce the case in which we replace the real mechanism with the ideal one inside S: $S[M^\star]$. The question above can now be translated to: *is it possible to derive the security of* $S[M]$ *from those of* M *and* $S[M^\star]$*?*

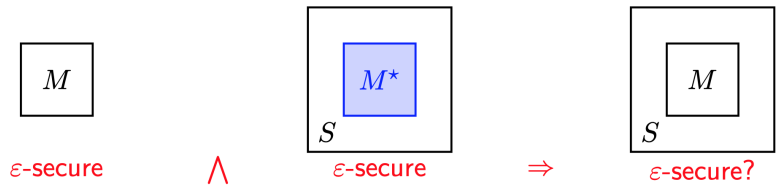


Figure 2.4: The composability question.

The approach we used before does not allow us to answer positively yet. We **must introduce a new metric** which does not rely on attacks success rate but rather on the intrinsic properties of the system. To work in this direction, let us introduce a new framework, in which we make use of a tool for common situations in security (e.g., intrusion detection, authenticity verification, etc.).

Definition 2.8 (Distinguisher)**.** With respect to Fig. 2.5, a **binary distinguisher** between two random variables x_0 and x_1 is a system D that is allowed to observe a realization of y without knowing in advance if $b = 0$ or $b = 1$ and should then guess which one holds. In general:

— x_i $i = 0, 1$ is characterized by its **probability mass densities** (PMDs) p_{x_i};

— D is composed of a decision function $g : \mathcal{Y} \mapsto \{0, 1\}$, i.e. $\hat{b} = g(y)$.

With respect to the scenario in Fig. 2.5, two random (binary) variables x_0, x_1 get a value according to their respective probability distributions. At this point, a **selector** b decides (in a random or deterministic way) which of these two values to assign to a realization y, which is then fed to the distinguisher module D whose goal is to

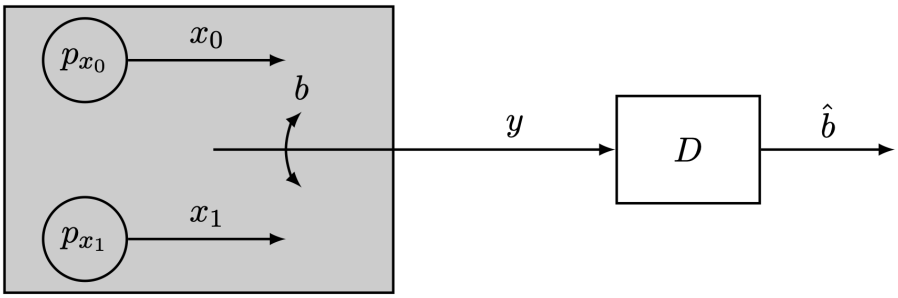


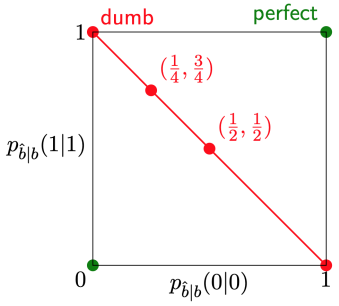
Figure 2.5: A distinguisher schema.

determine the index $\hat{b}$ (0 or 1) of the original x used to create y.
We need to introduce a metric to evaluate the quality of the distinguisher performances. The most straightforward way to do it is to consider the pair of **correct decision** probabilities $\left(p_{\hat{b}|b}(0|0), p_{\hat{b}|b}(1|1)\right)$ or equivalently the pair of **wrong decisions** $\left(p_{\hat{b}|b}(0|1), p_{\hat{b}|b}(1|0)\right)$.

Definition 2.9 (Distinguishability)**.** We define the **distinguishability** between x_0 and x_1 with D ($p_{\hat{b}|b}(\cdot|\cdot) \equiv p(\cdot|\cdot)$ to ease notation) as:

$$\begin{aligned} d_D(x_0, x_1) &= |p(0|0) - p(0|1)| = |p(1|1) - p(1|0)| \underset{(a)}{=} \\ &= |p(1|1) + p(0|0) - 1| \underset{(b)}{=} |1 - p(1|0) - p(0|1)| \end{aligned}$$

Where in (a) we used the fact that $p(1|0) + p(0|0) = 1$ and in (b) that $p(1|1) = 1 - p(0|1)$. For a perfect distinguisher $d_D(x_0, x_1) = 1$ and for a dumb distinguisher $d_D(x_0, x_1) = 0$.



The figure on the left depicts a summary of a distinguisher performances by comparing on a plane all possible pairs of right detections of 0 and 1. As we can notice, performances are excellent if all classifications are right (top-right corner) or all wrong (bottom-left corner, in this case it is sufficient to flip all predictions), while the module is considered dumb if its performance lays on the anti-diagonal of the square (random guess of $\hat{b}$).

Definition 2.10 (Unconditional indistinguishability)**.** Two variables x_0 and x_1 are said to be ε-unconditionally indistiguishable if, for any distinguisher D, it is $d_D(x_0, x_1) \leq \varepsilon$.

It is not always possible to find a perfect or even a good distinguisher, mainly because the two variable may not be perfectly distinguishable (see Fig. 2.6). Unconditional distinguishability is then a measure of "statistical distance" between two variables, which tells us how well a distinguisher can aim to perform.

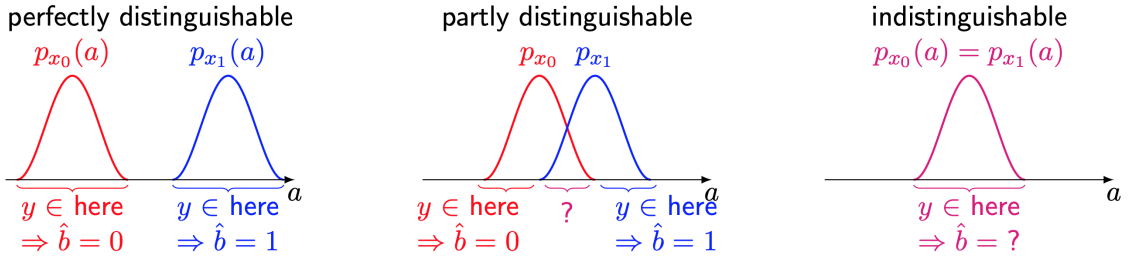


Figure 2.6: Indistinguishability of random variables

Definition 2.11 (Variational statistical distance)**.** The variational distance between two random variables x, y with alphabet $\mathcal{A}$ is defined as:

— For x, y **discrete**:

$$d_V(x, y) = \frac{1}{2} \sum_{a \in \mathcal{A}} |p_x(a) - p_y(a)|$$

— For x, y **continuous**:

$$d_V(x, y) = \frac{1}{2} \int_{\mathcal{A}} |p_x(a) - p_y(a)| \, \mathrm{d}a$$

that is, half the 1-norm distance between their PMDs.

The variational distance is an useful quantity that **can be exploited to measure distinguishability**. In fact, it is straightforward to prove the following result.

Proposition 2.1. Given a distinguisher D and two random variables x_0, x_1 with an alphabet $\mathcal{A}$ it holds:

$$\sup_D d_D(x_0, x_1) = d_V(x_0, x_1),$$

from which the following:

Corollary 2.1. Given two random variables x_0, x_1, $\forall \varepsilon > 0$ it holds:

$$d_V(x_0, x_1) \leq \varepsilon \iff x_0 \text{ and } x_1 \text{ are } \varepsilon\text{-distinguishable}$$

Sometimes, even if a good distinguisher exists in principle, it may not be possible to find a computationally efficient one. Hence, we use a real-world concrete **computational** definitions:

Definition 2.12 (Computational indistiguishability)**.** Two random variables x_0 and x_1 are said to be (ε, T_0)-computationally indistiguishable if, for any distinguisher D with complexity $T_D \leq T_0$, it holds $d_D(x_0, x_1) \leq \varepsilon$.

Definition 2.13 (Asymptotic computational indistiguishability)**.** Two sequences of random variables (or vectors) $x_{0,n}$ and $x_{1,n}$ are said to be computationally indistiguishable in the asymptotic formulation if, for any polynomials $p(\cdot), q(\cdot)$ and any sequence of distinguishers D_n with complexity $T_{D_n} \leq p(n)$, there exists n_0 such that $d_{D_n}(x_{0,n}, x_{1,n}) \leq 1/q(n)$, $\forall n > n_0$.

We can no2 generalize what said before for random variables in order to define distinguishability for quantum systems.

Definition 2.14 (Quantum distinguisher)**.** With respect to Fig. 2.7, a distinguisher between two quantum states $\hat{\rho}_0$ and $\hat{\rho}_1$ is a **binary measurement operation** $\left(\hat{M}_0, \hat{M}_1\right)$ on $\hat{\rho}_b$ without knowing in advance if $b = 0$ or $b = 1$ that outputs the result $\hat{b}$.

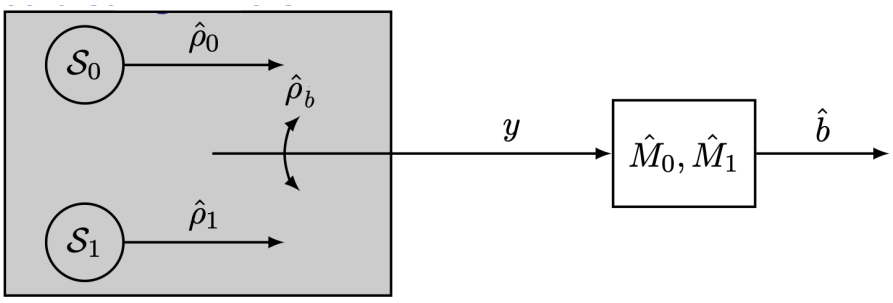


Figure 2.7: A quantum distinguisher schema

We can recognize the following:

— **Distinguisher probabilities:** $P_{i|j} = \mathrm{Tr}\left(\hat{M}_i^\dagger \hat{M}_i \hat{\rho}_j\right)$, $i, j = 0, 1$.

— **Distinguisher advantage:** $d_{\hat{M}}(\hat{\rho}_0, \hat{\rho}_1) = \left|P_{0|0} - P_{0|1}\right| = \left|\mathrm{Tr}\left(\hat{M}_0^\dagger \hat{M}_0 (\hat{\rho}_0 - \hat{\rho}_1)\right)\right|$.

— **Distinguishability:** $d(\hat{\rho}_0, \hat{\rho}_1) = \sup_M d_{\hat{M}}(\hat{\rho}_0, \hat{\rho}_1) = \frac{1}{2}\mathrm{Tr}(|(\hat{\rho}_0 - \hat{\rho}_1)|)$.

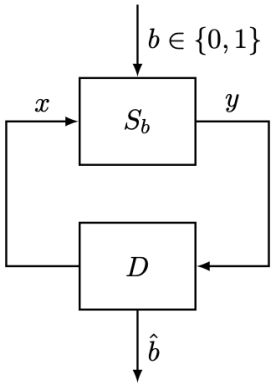


Figure 2.8

We can also use the definitions above to talk about **system distinguishers**. Looking at Fig. 2.8, a distinguisher between two probabilistic systems S_0 (characterized by the conditional PMD $p_{y_0|x_0}$) and S_1 (characterized by the conditional PMD $p_{y_1|x_1}$) is a third system D that is allowed to interact with a system S_b without knowing in advance if $b = 0$ of $b = 1$ and:

— can feed any input x to S_b;

— can observe the corresponding output y;

— should then guess whether $b = 0$ or $b = 1$.

D is composed of:

— an input selection strategy p_x (possibily adaptive, $p_{x|y}$);

— a decision function $g : \mathcal{X} \times \mathcal{Y} \mapsto \{0,1\}$, i.e. $\hat{b} = g(x,y)$.

Hence, generalization is straightforward.

Definition 2.15 (Indistinguishability of systems)**.** Two systems S_0 and S_1 are said to be ε-unconditionally indistiguishable if, for any distinguisher D, it holds $d_D(S_0, S_1) \leq \varepsilon$.

Definition 2.16 (Computational indistinguishability of systems)**.** Two systems S_0 and S_1 are said to be (ε, T_0)-computationally indistiguishable if, for any distinguisher D with complexity $T_D \leq T_0$, it holds $d_D(S_0, S_1) \leq \varepsilon$.

Definition 2.17 (Asymptotic computational indistinguishability of systems)**.** Two sequences of systems $S_{0,n}$ and $S_{1,n}$ are said to be computationally indistiguishable in the asymptotic formulation if, for any polynomials $p(\cdot)$, $q(\cdot)$ and any sequence of distinguishers D_n with complexity $T_{D_n} \leq p(n)$, there exists n_0 such that $d_{D_n}(S_{0,n}, S_{1,n}) \leq 1/q(n) \;\; \forall n > n_0$.

We can now define the security measure for a mechanism M in terms of indistinguishability from its ideal counterpart $M^\star$.

Definition 2.18 (Unconditional security)**.** A mechanism M is said to be ε-unconditionally secure if it is ε-unconditionally indistiguishable from its ideal counterpart $M^\star$.

Definition 2.19 (Computational security)**.** A mechanism M is said to be (ε, T_0)-computationally secure if it is (ε, T_0)-computationally indistiguishable from its ideal counterpart $M^\star$.

Definition 2.20 (Asymptotic computational security)**.** A sequence of mechanisms $\{M_n\}$ is said to be computationally secure in the asymptotic formulation if it is computationally indistiguishable from its ideal counterpart $M_n^\star$ in the asymptotic formulation.

In particular, in this framework we can exploit the following result.

Proposition 2.2 (Composability)**.** If a mechanism M is δ-unconditionally secure and its ideal counterpart $M^\star$ offers ε-unconditional security against a class $\mathcal{A}$ of attacks, then M offers $(\varepsilon + \delta)$-unconditional security against the same class $\mathcal{A}$.

Proof. Since $d(M, M^\star) \leq \delta$, there exist a joint conditional distribution of the outputs $p_{yy^\star|x}$ such that $P[y \neq y^\star | x = a] \leq \delta$, $\forall a \in \mathcal{A}_x$. Therefore, for all $A \in \mathcal{A}$, and by the total probability theorem:

$$\begin{aligned} P[S_\mathcal{A}; A, M] &= P[S_\mathcal{A} | y = y^\star; A, M] \, P[y = y^\star; A, M] + \\ &+ P[S_\mathcal{A} | y \neq y^\star; A, M] \, P[y \neq y^\star; A, M] \\ &\leq P[S_\mathcal{A}; A, M^\star] \cdot 1 + 1 \cdot P[y \neq y^\star; A, M] \\ &\leq \varepsilon + \delta \end{aligned}$$

□

Similar results can be proved for the computational and asymptotic computational cases.
Going back to the computational complexity classes framework, this translates in the following:

Asymptotic computational security $\iff$ distinguishing $\{M_n\}$ from $\{M_n^\star\} \notin$ BPP

Idem vs a quantum adversary $\iff$ distinguishing $\{M_n\}$ from $\{M_n^\star\} \notin$ BQP

Chapter 3

Entanglement

1 Bipartite systems

Up to this point in the course we have been dealing with just one Hilbert space, but the protocols we are going to learn exploit systems living in larger Hilbert spaces. How can I define a joint state for both systems? The answer lies in the **tensor product** $\otimes$ operation. We consider a bipartite system, composed of two spaces $\mathcal{H}_A$

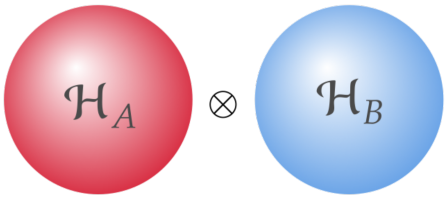


Figure 3.1: A bipartite system.

and $\mathcal{H}_B$, each one with its own basis $\{|a_i\rangle_A\}_{i=0}^{d_A-1}$, $\{|b_i\rangle_B\}_{i=0}^{d_B-1}$. A quantum state (ket or bra) in one of the two spaces will take the form of Eq. (1.1).
A quantum state of the bipartite system $\mathcal{H}_S = \mathcal{H}_A \otimes \mathcal{H}_B$ is instead written as follows:

$$|\psi\rangle_{AB} = \sum_a \sum_b \psi_{ab} |a\rangle_A |b\rangle_B \tag{3.1}$$

where the basis of the tensor product space can be written in many ways: $|a\rangle_A \otimes |b\rangle_B \equiv |a\rangle_A |b\rangle_B \equiv |a, b\rangle_{AB}$. The final joint space will have dimension $d_A \cdot d_B$.
Only states of the same space "interact" with each other, e.g. a state in A will interact only with states in A,

$$\langle a', b'|a, b\rangle = \langle a|a'\rangle_A \cdot \langle b|b'\rangle_B$$

Similarly, an operator defined on the tensor product space $\mathcal{H}_S$ acts in the two spaces separately; given $|\Psi\rangle_{AB} = \sum_{ab} \psi_{ab} |a\rangle_A |b\rangle_B$ and two operators $\hat{A}$ and $\hat{B}$ that act as

$\hat{A}\,|a\rangle_A = |\alpha_a\rangle$ and $\hat{B}\,|b\rangle_B = |\beta_b\rangle$,

$$\hat{A}\otimes\hat{B}\,|\Psi\rangle_{AB} = \sum_{a,b}\psi_{ab}\,|\alpha_a\rangle\,|\beta_b\rangle \tag{3.2}$$

2 Entangled states

2.1 Pure states

Definition 2.1 (Separable state)**.** Given two Hilbert spaces $\mathcal{H}_A$ and $\mathcal{H}_B$, a quantum state of the bipartite system $\mathcal{H}_A\otimes\mathcal{H}_B$ is said to be **separable** if it can be written as

$$|S\rangle_{AB} = |\psi\rangle_A\otimes|\psi\rangle_B \tag{3.3}$$

Definition 2.2 (Entangled state)**.** Given two Hilbert spaces $\mathcal{H}_A$ and $\mathcal{H}_B$, a quantum state of $\mathcal{H}_A\otimes\mathcal{H}_B$ is said to be **entangled** if it can **not** be written as Eq. (3.3), i.e. a tensor porduct of two (or more) different states,

$$|S\rangle_{AB} \neq |\psi\rangle_A\otimes|\psi\rangle_B \tag{3.4}$$

How can I recognize whether a system is entangled or not? In some simple cases the difference is immediate. For example, $|\psi\rangle = |0\rangle_A\,|0\rangle_B$ is separable, and $|\psi^+\rangle = \frac{1}{\sqrt{2}}(|0\rangle_A\,|1\rangle_B + |1\rangle_A\,|0\rangle_B)$ is entangled, but it can be much harder with more complex state to perform this task. We need something more deterministic to do it, and a result from linear algebra comes into help.

Theorem 2.1 (Schmidt decomposition)**.** Let $\mathcal{H}_A$ and $\mathcal{H}_B$ be two Hilbert spaces of dimensions n and m respectively. Let $k \leq \min\{m,n\}$. For any vector $|\psi\rangle_{AB}$ in the tensor product space $\mathcal{H}_A\otimes\mathcal{H}_B$, there exist a set of orthonormal basis $\{v_1,\ldots,v_n\}\in\mathcal{H}_A$ and $\{w_1,\ldots,w_m\}\in\mathcal{H}_B$ such that

$$|\psi\rangle_{AB} = \sum_{i=1}^{k}\sqrt{\lambda_i}\,|v_i\rangle_A\,|w_i\rangle_B \qquad\qquad \sum_{i=1}^{k}\lambda_i = 1 \tag{3.5}$$

where the scalars λ_i are real, non-negative and unique up to re-ordering.

Proof. Omitted. □

Note that in Eq. (3.5) we are summing over indices of only one basis. Schmidt decomposition is thus simpler, in a sense, than the usual decomposition of $|\psi\rangle$ in two *generic* basis $\{\,|\alpha\rangle_A\}$ and $\{\,|\beta\rangle_B\}$ of $\mathcal{H}_A$ and $\mathcal{H}_B$,

$$|\psi\rangle = \sum_{ij}^{\dim\mathcal{H}_{A,B}}\gamma_{ij}\,|\alpha_i\rangle_A\,|\beta_j\rangle_B$$

where the sum is over *both* indices i and j. The result from Theorem 2.1 shows how it is possible to remove one of the two indices with an appropriate choice of basis.

Let's now consider a vector in the tensor product space of two Hilbert spaces in the form of a Schmidt decomposition $|\psi\rangle_{AB}$

$$|\psi\rangle_{AB} = \sum_{i=1}^{k} \sqrt{\lambda_i} \, |v_i\rangle_A \, |w_i\rangle_B$$

and build the density matrix $\rho = |\psi\rangle \langle\psi|$. Then, if we take the partial trace of ρ with respect to either system A or B

$$\rho_1 = \underset{2}{\mathrm{Tr}} \, |\psi\rangle \langle\psi| = \sum_{i=1}^{k} \lambda_i \, |v_i\rangle \langle v_i|$$

and symmetrically

$$\rho_2 = \underset{1}{\mathrm{Tr}} \, |\psi\rangle \langle\psi| = \sum_{j=1}^{k} \lambda_j \, |w_j\rangle \langle w_j|$$

i.e. a diagonal matrix whose non-zero entries are $|\sqrt{\lambda_i}|^2 = \lambda_i$. In other words, the Schmidt decomposition shows that the reduced states of ρ on either subsystems have the same spectrum.

Definition 2.3 (Schmidt coefficients)**.** The strictly positive values λ_i in the Schmidt decomposition of $|\psi\rangle_{AB}$ are its **Schmidt coefficients**.

Definition 2.4 (Schmidt rank)**.** The number k of Schmidt coefficients λ_i is called the **Schmidt rank**.

The definition of the Schmidt rank is useful to understand if a **pure state** is entangled or not. In particular it holds

$$k = 1 \iff |\psi\rangle \text{ is } \textbf{not entangled} \text{ (i.e. separable).} \tag{3.6}$$

On the other hand, if the Schmidt rank is strictly greater than one ($k \geq 1$) then it means that the state is instead entangled.
Since the coefficients of the Schmidt decomposition are the diagonal elements of the partial trace of the density matrix of a bipartite system with respect to one of the two subsystems, the condition to detect entanglement boils down to verify whether $\mathrm{rank}(\rho_A) > 1$. If so, ρ_{AB} is entangled, otherwise it is separable.
k, being a natural number, only informs about the presence or absence of correlations, but does not quantify their "intensity". In practice, in order to use entanglement in computations, it is necessary to to work with strong correlations, and

$k = 1$ is not necessarily an indication of an experimentally "usable" state, so it will be necessary to evaluate more sophisticated parameters. It is important to stress once again that this is valid **only for pure states**.

The definition of entanglement generalises also in a multipartite system scenario: in this case we could calculate the density matrix for just a couple of systems simply by tracing out all the useless information on the systems we are not interested in.

EXAMPLE. For a two-qubits quantum system we can define some **maximally entangled** states, also known as **Bell states**

$$|\psi\rangle_{AB}^{\pm} = \frac{1}{\sqrt{2}}(|01\rangle_{AB} \pm |10\rangle_{AB})$$

$$|\phi\rangle_{AB}^{\pm} = \frac{1}{\sqrt{2}}(|00\rangle_{AB} \pm |11\rangle_{AB})$$

On the other hand, a not maximally entangled state is in the form

$$\cos\theta\,|00\rangle + \sin\theta\,|11\rangle$$

which is also the most general expression for a 2-qubits system. Moreover, it coincides with its Schmidt decomposition (with at most two terms): indeed the sum of its coefficients is equal to 1. By changing the parameter θ one can change the level of entanglement.

Correlations in an entangled state cannot be explained classically. Let's consider the two Pauli matrices as operators $\hat{\sigma}_z$ and $\hat{\sigma}_x$. Their action corresponds to the projective measure in the basis states, given respectively by $\{|0\rangle, |1\rangle\}_z$ and $\{|+\rangle, |-\rangle\}_x$.
If we consider the expectation values of $\sigma^A \otimes \sigma^B$ over the Bell state $|\psi^-\rangle$,

$$\langle\psi_{AB}^-|\,\sigma_z^A \otimes \sigma_z^B\,|\psi_{AB}^-\rangle = -1 \qquad\qquad \langle\psi_{AB}^-|\,\sigma_x^A \otimes \sigma_x^B\,|\psi_{AB}^-\rangle = -1$$

meaning that they are maximally anti-correlated. But this is surprising! Indeed, if we take the same expectation value on a single element of the total Bell state we have $\langle 01|\,\sigma_x^A \otimes \sigma_x^B\,|01\rangle = 0$ immediately, since σ_x^B acts on both qubits individually like a NOT gate and then there is a Dirac product between two orthogonal quantum states. With such a state, if we measure $|0\rangle$ in the system A, then the state counterpart in B collapses immediately to $|1\rangle$. This is to show that entanglement is a correlation which is **not pre-present like in the classical case**. In fact, when we calculate the expected value over the single constituents of the Bell state we get no correlation: entanglement is a correlation which can not be explained via a simple correlation in the original sub-components but that instead **emerges only in their quantum superposition**. To verify that, let's consider a maximally separable state,

$$|\rho\rangle = \frac{1}{2}\,|01\rangle\,\langle 01| + \frac{1}{2}\,|10\rangle\,\langle 10|$$

i.e. an incoherent superposition of 0 and 1. The same calculation saw above yield to no correlation for both the single element and the complete state.

To conclude, entanglement is much more than classical correlation: another way to see it is that of a correlation among different basis, a behaviour that no classical system can reproduce.

2.2 Mixed states

In this section we will consider the case of entanglement for mixed state for bipartite systems: the generalisation to multi-partite follows directly. Let's consider a separable mixed state

$$\hat{\rho}_{AB} = \sum_{n=1}^{N} p_n \, \hat{\rho}_n^A \otimes \hat{\rho}_n^B$$

This state can be prepared using only **LOCC** - **l**ocal **o**perations and **c**lassical **c**ommunication. Local means acting only on one subsystem, and classical communication involves two parties A and B talking to each other to define together how each matrix element is prepared and with which probability.
A state like this one is classically correlated, and for this purpose we notice how the index n is the same in the two density matrices. Without classical correlations the state should have been

$$\hat{\rho}_{AB} = \sum_{n=1}^{N} \sum_{m=1}^{M} p_n^A p_m^B \, \hat{\rho}_n^A \otimes \hat{\rho}_m^B \tag{3.7}$$

Definition 2.5 (Entangled Mixed State)**.** A mixed state that cannot be written as in Eq. (3.7) is said to be **entangled**.

LOCC maintain the level of entanglement between states: it follows directly that LOCC cannot be used to create entangled states. To create entanglement I need **global operations**. Bell states belong to the same entanglement class because they are LOCC connected.

Definition 2.6 (Partial transpose)**.** If we have a general state ρ which acts on $\mathcal{H}_A \otimes \mathcal{H}_B$ as

$$\rho = \sum_{ijkl} p_{kl}^{ij} |i\rangle\langle j| \otimes |k\rangle\langle l|$$

its partial transpose (with respect to the B party) is defined as

$$\rho^{T_B} \coloneqq (I \otimes T)(\rho) = \sum_{ijkl} p_{kl}^{ij} |i\rangle\langle j| \otimes (|k\rangle\langle l|)^T \tag{3.8}$$

$$= \sum_{ijkl} p_{kl}^{ij} |i\rangle\langle j| \otimes |l\rangle\langle k| \underset{(a)}{=} \sum_{ijkl} p_{lk}^{ij} |i\rangle\langle j| \otimes |k\rangle\langle l| \tag{3.9}$$

where in (a) we used the fact that when a transposition is done, it is commonly used to exchange directly the vectors, but it is the same to swap simply the coefficients (for a partial transposition on the B party).
Note that the partial in the name implies that only part of the state is transposed. More precisely, $(I \otimes T)(\rho)$ is the identity map applied to the A party and the transposition map applied to the B party.

Theorem 2.2 (Peres–Horodecki criterion)**.** If we have a general density matrix $\hat{\rho}_{AB}$ that acts on the tensor product space,

$$\hat{\rho}_{AB}^{T_A} < 0 \Rightarrow \hat{\rho}_{AB} \text{ is entangled} \tag{3.10}$$

Proof. It's easier to demonstrate that a separable mixed state must have positive eigenvalues (logically, the thesis of theorem will gently follow). Suppose the state is separable, then it can be written as

$$\hat{\rho}_{AB} = \sum_{n=1}^{N} p_n \, \hat{\rho}_n^A \otimes \hat{\rho}_n^B$$

and the partial transpose operation acts simply as follows

$$\hat{\rho}_{AB}^{T_A} = (T_A \otimes I)\hat{\rho}_{AB} = \sum_{n=1}^{N} p_n \, (\hat{\rho}_n^A)^T \otimes \hat{\rho}_n^B$$

In this situation, since the transposition map (the partial transposition on a separable state is a full transposition on one of the systems) **preserves eigenvalues**, the spectrum of $(\hat{\rho}_n^A)^T$ is the same as the one of $\hat{\rho}_n^A$, and in particular $(\hat{\rho}_n^A)^T$ must still be semidefinite positive. As a consequence, also $\hat{\rho}_{AB}^{T_A}$ is semidefinite positive. □

The criterion states that if $\hat{\rho}_{AB}^{T_A}$ has a negative eigenvalue, $\hat{\rho}_{AB}$ is entangled. In other words, if $\hat{\rho}_{AB}$ is separable then all the eigenvalues of $\hat{\rho}_{AB}^{T_A}$ are non-negative. For this reason, this result is also known as **PPT** criterion for positive partial transpose. This is coherent with the fact that the transpose operation is **not a completely positive operator** (i.e. it has no physical meaning) and thus the partial transpose can lead to negative results.
The definition is independent on the system we do the partial trace on.
If we are dealing with a product space of dimension 2×2 (qubits) and 2×3 (*qutrits*), the opposite way is also valid. This holds for bi-partite states only.

Bound entanglement

Definition 2.7 (Entanglement distillation)**.** Entanglement distillation (also called **entanglement purification**) is the transformation of N copies of an arbitrary

entangled state ρ into some number of approximately pure Bell pairs, using only LOCC.

Quantum entanglement distillation can in this way overcome the degenerative influence of noisy quantum channels by transforming previously shared less entangled pairs into a smaller number of maximally entangled pairs.

Entangled quantum states can be classified into two types: those that can be distilled into pure entangled states using LOCC and those that cannot. Undistillable entangled states are called **bound entangled**. This class of states are a weak form of quantum entanglement and are **not** detectable by the PPT criterion and thus other entanglement criteria are needed for their detection. Indeed, bipartite entangled states that have a non-negative partial transpose ($\hat{\rho}_{AB}^{T_A} \geq 0$) are **all** bound-entangled.
There are also multipartite entangled states that have a negative partial transpose with respect to some bipartitions, while they have a positive partial transpose to the other partitions, nevertheless, they are undistillable.

The possible existence of bipartite bound entangled states with a negative partial transpose is still under intensive study.

Monogamy of entanglement

In quantum physics, **monogamy** describes the fundamental principle that quantum entanglement cannot be freely shared between arbitrarily many parties. According to monogamy, in order for two qubits $|\psi\rangle$ and $|\phi\rangle$ to be maximally entangled, they must not be entangled with any third qubit $|\chi\rangle$ whatsoever. Even if $|\psi\rangle$ and $|\phi\rangle$ are not maximally entangled, the degree of entanglement between them constrains the degree to which $|\psi\rangle$ can be entangled with $|\chi\rangle$. This principle is of course relevant for Quantum Key Distribution (QKD) as we will see during these lectures. In particular, if $|\chi\rangle$ shares some entanglement with $|\psi\rangle$ and $|\phi\rangle$ then there will be less entanglement between the last two, and so we have a way to feel the presence of an **eavesdropper** and thus bound the information leaked.

Chapter 4

Quantum Random Number Generators (QRNGs)

Many things in this chapter are taken from [Herrero-Collantes and Garcia-Escartin(2017)], a comprehensive review paper also useful to dig deeper into the subject.

Why Random Numbers?

The first question that may arise when starting this chapter is *"Why do we need truly random numbers?"*. Indeed, the generation of good random numbers (RN) impacts basic research and other scenarios beyond pure academic interests: RN are required for **countless applications**, such as simulations and cryptography - given the fact that any cryptographic primitive, classical or quantum, requires good RN generators. For most applications, it is of paramount importance to know if a set of numbers is truly random, pseudo-random or contains some residual correlations: we should avoid to have smart players in our Casino that are able to predict the output of a slot machine. Nevertheless, a vast majority of simulations, games, or security protocols still rely on **classical randomness**, that in turn depends on the **ignorance on the initial condition** of a specific algorithm. Randomness is thus not intrinsic in the physical process (as in QM), where we can only predict the **probability** of a certain outcome and not the **outcome itself**.
But how do we usually generate RN?

1 Pseudo Random Number Generators (PRNGs)

The classical randomness can be generated with simple deterministic algorithms implemented by software, or exploiting the randomness present in some physical processes (such as thermal fluctuations of a resistor). This methods allow to generate a string of data that **resembles** a casual sequence of numbers (Fig. 4.1). Again, all of these processes are **not based on truly and intrinsic randomness**. We

will now discuss some of the most (in)famous examples of PRNGs.

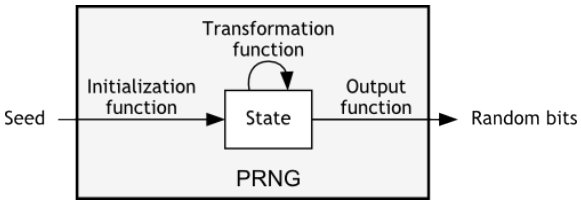


Figure 4.1: Simple schema of a PRNG.

Example (**Linear Congruential Generator, LCG**). Given some parameters a, b, a seed x_0 and a fixed parameter N (could be also very long), we draw the $n+1$-th random number in the following way

$$x_{n+1} = (ax_n + b) \mod N \tag{4.1}$$

Here, the modulus operation breaks the linearity and the period is at most N. The properties of the output depend heavily on the correct choice of these parameters. A poor choice can create an output sequence with a short period. If we fix for example $a = 2^{16} + 3 = 65539$, $b = 0$ and $N = 2^{31}$ we are talking of the infamous RANDU algorithm. But why is it so bad at creating randomness? Let's consider for example the generation of x_{n+2}

$$\begin{aligned} x_{n+2} &= a^2 x_n && \mod 2^{31} \\ &= (2^{16}+3)^2 x_n && \mod 2^{31} \\ &= (2^{32} + 6 \cdot 2^{16} + 9) x_n && \mod 2^{31} \\ &\underset{(a)}{=} [6 \cdot (2^{16}+3) - 9] x_n && \mod 2^{31} \\ &= 6x_{n+1} - 9x_n && \mod 2^{31} \end{aligned}$$

where in (a) we remember that $2^{32} \mod 2^{31}$ is 0. So, given two consequent numbers x_n and x_{n+1} we are able to predict all the numbers that are going to come out from RANDU.

Example (**Blum Blum Shub, BBS**). In this case, the output bits come from the recursive formula

$$x_{n+1} = x_n^2 \mod N$$

for $N = p \cdot q$ the product of two primes. This is a quadratic PRNG, harder to crack than a linear one, but can be demonstrated that if we are able to find the two factors p and q the security of the algorithm drops dazzlingly. Breaking BBS is equivalent to factoring: although this is considered computationally secure in many

cryptographic protocols, an attacker with a quantum computer that knows N could use Shor's algorithm for integer factorization to break the security of the generator.
EXAMPLE (**Mersenne Twister**). This is the number generator implemented in Excel and in Python and in its standard implementation (MT19937) it has a period $\mathcal{P} = 2^{19937} - 1$ (Mersenne prime). It has been proved that it can be hacked just by knowing 624 consequent outputs.[1].

From a RNG we require not only for the numbers to be uniformly distributed, namely that $P(x_n) = 0.5$, but we need completely conditional independence on the previous outcomes of the generator, $P(x_n|x_{n-1}, \ldots, x_1) = 0.5$ and no classical random generator is able to provide this. This particular property is called **backward security**: this security guarantees that the encrypted messages in a session should remain secure even if "backward" long-term key corruption occurs. It is hard to prove if a sequence of numbers is truly random or not, and we do it by means of **random tests**. However, these are only a sufficient condition but not a necessary one: if a test doesn't find correlations between numbers in a line, it doesn't mean that the sequence is perfectly random.
We have shown how PRNG have **low security**: one of their perks however is that a pseudo-random algorithm shows **high efficiency**, meaning that a long string of random bits can be generated in a small amount of time. Up to date, we do not know if there is a way to generate true random number through classical processes. We first generate a string of numbers that cannot be completely random and than by post-processing this we extract real randomness.

Fig. 4.2 shows the block diagram of a typical physical random number generator. The two most important blocks in which the random number generator can be

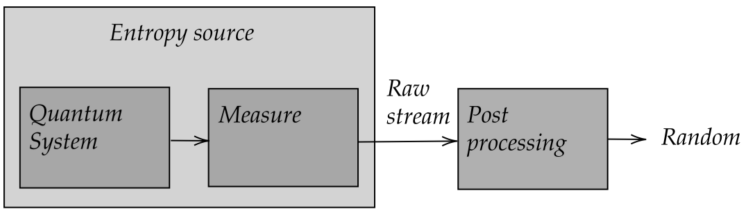


Figure 4.2: Block description of a physical random number generator.

subdivided into are the **entropy source** and the **post-processing stage**. The entropy source consists of a physical system with some random physical quantity and the measurement equipment that reads these random variables. Measurement and quantization are noisy processes and there will be some contamination in what

[1]For a practical implementation see here

is called the raw bit string - even if the measured quantity is truly random and free from correlations. The postprocessing block takes the raw bits and distills a shorter sequence without correlations. The most important phase in postprocessing is randomness extraction. Randomness extractors are functions that transform the bits from the raw sequence into a uniform random sequence at the output with most or all of the randomness available in the input.
In our following discussion, we concentrate on the different **quantum systems** that can work as an entropy source. But first, *how can we conveniently measure randomness?*

2 Entropy estimation

An important quantity that springs from Physics is entropy, and in its many forms offers a convenient way to **measure randomness**. The different entropies give a mathematical measure for **surprise** (how unexpected a value is). We express entropy in bits in the information theory sense. We now define the first simple entropy measurement:

Definition 2.1 (Shannon entropy)**.** For a random variable X with probability distribution P_X, so that $P_X(x)$ is the probability of getting the outcome X that takes value in the alphabet $\mathcal{A}$ with cardinality $|\mathcal{A}| = N$, the Shannon entropy of X, $H(X)$, is defined as

$$H(X) = -\sum_{x\in\mathcal{A}} P_X(x)\log_2 P_X(x) \tag{4.2}$$

The Shannon entropy gives the average number of bits of information we can extract from a single outcome. For an alphabet with a uniform probability distribution, all the results are equally likely and we need $\log_2 N$ bits to describe them. Shannon entropy offers a rough estimation of randomness. Ideally, we would like to generate an almost uniform distribution with a Shannon entropy as close to $\log_2 N$ as possible. A higher Shannon entropy means we have a distribution closer to uniform and that we can extract more random bits from the process, but there are other entropy measures that can give us a more useful figure when deciding how to use a randomness extractor to make the most efficient use of the available randomness.
However, the Shannon definition of entropy applies only to i.i.d. scenarios (an example could be a sequence of coin tosses, where one is independent from the other), for very long sequences of extractions. However, we would like to characterize also **single-shot** operations, related to a single run of an experiment, i.e. a single generation/extraction rather than the full stream of generations.
It is thus necessary to introduce other forms of entropy: a generalization of Shannon entropy is the family of **Rényi entropies** of order α, defined as

$$H_\alpha(x) = \frac{1}{1-\alpha}\log_2 \sum_{x\in\mathcal{A}} P_X(x)^\alpha \tag{4.3}$$

where the Shannon entropy corresponds to the Rényi entropy in the limit $\alpha \to 1$ (apply de l'Hôpital rule).
A property that holds is that, for any distribution, $H_\alpha(X) \geq H_\beta(X)$ for $\alpha \leq \beta$. A particularly useful quantity is the **min-entropy** $H_\infty(X)$, which comes from taking the Rényi entropy when $\alpha \to \infty$. The min-entropy gives a lower, worst-case bound[2] to all the Rényi entropies; another way to see it is as the number of random bits one could retrieve from a certain process. The name min-entropy stems from the fact that it is the smallest entropy measure in the family of Rényi entropies. In this sense, it is the strongest way to measure the information content of a discrete random variable. In particular, the min-entropy is never larger than the Shannon entropy.
It is defined as the negative logarithm of the probability of the most likely outcome:

$$H_\infty(x) \equiv H_{\min}(x) = -\log_2(\max_{x \in \mathcal{A}} P_x). \tag{4.4}$$

Consequently,

$$P_g = 2^{-H_\infty(X)} = 2^{-H_{min}(X)} \tag{4.5}$$

corresponds to the probability of guessing at the first attempt the outcome from measuring a random variable X with a known distribution. Put it in another way, if I have a random generator whose outputs have different probabilities ($x \neq x'$ implies $p_x \neq p'_x$) and I want to predict the next outcome on a single-shot measurement, I will (reasonably) guess the most probable one. This is why it is linked to and relevant for a single-shot operation.

We can generalize this expressions to a quantum system, substituting classical probabilities with the eigenvalues of a density matrix. The **Von Neumann** entropy is defined as

$$S(\hat{\rho}) = -\operatorname{Tr}[\hat{\rho} \log_2 \hat{\rho}]. \tag{4.6}$$

Rényi entropies, including Shannon entropy and min-entropy, can be **generalized to study joint distributions** where part of the system is in the power of a legitimate user Alice (A) and part of the system, which can be correlated to the first part, is in the possession of an eavesdropper Eve (E). This is important to understand what kind of information a system that is for example entangled with our system could share. In random number generation, the most useful quantity is the **(quantum) conditional min-entropy**

$$H_{\min}(A|E)_{\hat{\rho}_{AE}} = \sup_{\hat{\sigma}_E}[-\log_2 \lambda] \quad \lambda \in \mathbb{R} \text{ smallest number s.t. } (\lambda \mathbb{1}_A \otimes \hat{\sigma}_E - \hat{\rho}_{AE}) \geq 0 \tag{4.7}$$

[2]The min-entropy $H_\infty(X)$ is the **largest** real number such that all events can occur with probability **at most** $2^{H_\infty(X)}$.

of $\hat{\rho}_{AB}$ related to the density matrix of the **environment** $\hat{\sigma}_E$. This is a sort of double maximization: we change $\hat{\sigma}_E$, maximizing the quantity with respect to λ and then change again $\hat{\sigma}_E$. The quantum conditional min-entropy gives **how much information about the results of a measurement by** A **can be inferred from measurements on** E alone. For classical distributions, $2^{-H_\infty(A|E)_\rho}$ gives the probability of guessing the outcomes of A from our knowledge of E using the optimal strategy. If there is no side information (the systems of A and E are uncorrelated), we recover the definition and interpretation of the min-entropy in Eq. (4.4) (see also the explicit value of $H_{\min}$ calculated in Eq. (4.9)).
At the same time we can also define

$$H_{\max}(A|E)_{\hat{\rho}_{AE}} = -H_{\min}(A|B)_{|\psi\rangle_{ABE}} \tag{4.8}$$

where $|\psi\rangle_{ABE}$ is a purification of $\hat{\rho}_{AE}$, meaning that we have three systems and $|\psi\rangle_{ABE}$ is a pure state in an generic enlarged system that includes also a new portion B, $\hat{\rho}_{AE} = \mathrm{Tr}_B(|\psi\rangle_{ABE}\langle\psi|)$.
Let's give some explicit values in some particular cases of $H_{\max/\min}$.

— **No correlations**, $\hat{\rho}_{AE} = \hat{\rho}_A \otimes \hat{\rho_E}$:

$$H_{\min} = -\log_2(\max \lambda_A) \qquad H_{\max} = 2\log_2 \mathrm{Tr}\left(\sqrt{\hat{\rho}_A}\right) \equiv H_{1/2}(\hat{\rho}_A) \tag{4.9}$$

where λ_A are the eigenvalues of $\hat{\rho}_A$. With no correlations the classical result is retrieved.

— **Pure state**, $|\psi\rangle_{AE}$:

$$H_{\min} = -2\log_2 \mathrm{Tr}\left(\sqrt{\hat{\rho}_A}\right) \qquad H_{\max} = \log_2 \mathrm{Tr}(\max \lambda_A) \tag{4.10}$$

This expression follow directly from the definition in Eq. (4.8). For a maximally entangled state $H_{\min} = 0$ since the partial matrix $\hat{\rho}_A \equiv \mathbb{1}_A$[3] i.e. there is no residual information but rather full correlation between the environment and the system.

[3]Given two systems with the same dimension d, a maximally entangled state can we written as $|\psi\rangle_{AB} = \frac{1}{\sqrt{d}}\sum_{n=0}^{d-1}|v_n\rangle_A|w_n\rangle_B$ following Schmidt decomposition. Consequently

$$\hat{\rho}_A = \underset{B}{\mathrm{Tr}}\,(|\psi\rangle\langle\psi|) = \frac{1}{d}\sum_{k,k'}\sum_{n=0}^{d-1}\langle n|w_k\rangle_A\langle w_{k'}|n\rangle_B|v_k\rangle\langle v_{k'}| = \frac{1}{d}\sum_{k=0}^{d-1}|v_k\rangle\langle v_k| = \frac{\mathbb{1}_d}{d}. \tag{4.11}$$

So tracing out a component in a two component maximally entangled state leads to the identity i.e. the most uninformative state.

In our scenario the state $\hat{\rho}_{AE}$ is not generic because - as for any QRNG - we perform a measurement on the A system. After the measurement the state could always be written in the form

$$\hat{\rho}_{AE} \underset{\text{MEASURE}}{\longrightarrow} \hat{\rho}_{AE} = \sum_X P_X \left|X\right\rangle \left\langle X\right| \otimes \hat{\rho}_E^x \qquad \left\langle X|X'\right\rangle = \delta_{X,X'} \tag{4.12}$$

i.e. a **classical quantum density matrix**, since we have a classical part (the first term in the tensor product, where P_X is the probability output of the matrix) together with some quantum state of the environment. The superscript indicates the value of the classical system upon which the state depends. The difference between the two terms is that the first is diagonal while the second is (in general) not.[4]

Now, what is the probability $P_g(X|E)$ of getting the output X supposing that we have full control over the environment, i.e. *how can we use the knowledge of the environment to predict the outcome in A?* This result is given by

$$P_g(X|E) = \max_{\{\hat{F}_X\}} \sum_X P_X \operatorname*{Tr}_E (\hat{F}_X \hat{\rho}_X^E). \tag{4.13}$$

The set $\{\hat{F}_X\}$ is a general POVM over the environment E: the idea is to maximize the probability for E to have the same result as that in system A as output, i.e. the probability that the eavesdropper can successfully predict the output. This is taken into account by the term $\operatorname*{Tr}_E (\hat{F}_X \hat{\rho}_X^E)$: we then sum over all the possibilities weighted by their relative probability. When the state is not correlated, for instance has the form $\hat{\rho}_{AE} = (\sum_X P_X \left|X\right\rangle \left\langle X\right|) \otimes \hat{\rho}_X^E$ the maximization over $\hat{F}_X$ loses meaning and one retrieves the classical definition of min-entropy.
It can be shown [Konig *et al.*(2009)Konig, Renner, and Schaffner] that

$$P_g(X|E) = 2^{-H_{\min}(X|E)} \tag{4.14}$$

so we find a direct link between mean entropy and the guessing probability, which is the generalization of Eq. (4.5): in the separable case we retrieve the classical result as expected.

The problem however, is that for a general unknown source, estimating the min-entropy is far from trivial. The problem is intractable for any reasonable sampling circuit with limited size. We can only determine min-entropy from measurement inefficiently. Normally, physical random number generators use conservative, worst-case bounds for the min-entropy based on a deep analysis of the physical origin of the randomness.

[4] In other words, one could say that the second term has in general non-zero **coherences**, i.e. off-diagonal terms.

3 Quantum Random Number Generators (QRNGs)

We can define different notions of QRNG:

— **Trusted scenario**: We are assuming that the system is isolated and separable ($\hat{\rho}_{AE} = \hat{\rho}_A \otimes \hat{\rho}_E$, i.e. low correlation with the environment), that the state is pure and the measurement is a perfect measurement. As a consequence, all the measures of entropy reduces to the classical quantities.

— **(Semi-)device-indipendent**: In this case we do not have full control over the environment. This type of generators exploit some relation in the in-out system, bounding the (conditional-) min-entropy. In this case the security is higher, at the price of having a lower generation rate (usual trade-off). This is due to the fact that we impose less hypotheses on the source and/or measure, thus making the apparatus more robust to noise/imperfections.

The **trade-off** between security and ease of implementation often arises for RN generators, as will we see later on.
As before, the measure of randomness and consequently the goodness of the protocol will be the min-entropy $H_\infty(X)$ and the Shannon entropy $H(X)$ (for the particular cases of i.i.d. scenario in the long run).

3.1 Trusted QRNGs: discrete variables

For this section we focus on **discrete variables** (i.e. finite dimension Hilbert space): the typical example is the **photon detector** where the variable (the number of incident photons) is discrete.

Radioactive decay

Radioactive decay was a particularly accessible source of true randomness. For simplicity, most radioactivity-based quantum random number generators were based on the detection of β radiation (emitted electrons). The probability of any given atom to decay in a time interval $(t, t+dt)$ is given by an exponential random variable so that $P(t)dt = \lambda_m e^{-\lambda_m t} dt$ for a material with a decay constant λ_m. Under some mild assumptions the time between detected pulses is also an exponential random variable. The times are independent from previous results and the number of pulses that arrive in a fixed time period follows a **Poisson distribution** that can be used to generate RN.
This kind of QRNGs are the forerunners of the present day optical QRNGs that use similar concepts and circuits, but replace the radioactive source and the Geiger counter with photon sources and detector. Indeed, for a laser source the probability of detecting a given number of photons follows again a Poisson statistics.
Simple examples of RNGs based on radioactive decay are related to the notion

of **time of arrival** (TOA or ToA), i.e. the absolute time instant when a signal emanating from a transmitter reaches a remote receiver. The randomness in the time of arrival can be converted into random digits in a few different ways.

— **Fast-clock method**. Referring to Fig. 4.3: A fast clock (i.e. with a clock cycle frequency higher than the rate of detection, $\nu > \lambda$) (bottom line) is used to increase a counter. Whenever a detection is made (upper line), the counter is read and reset, generating a number whose parity constitutes a new random bit.

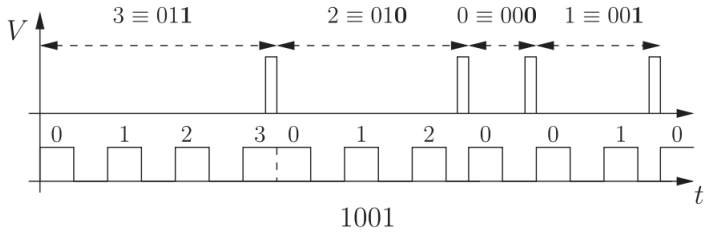


Figure 4.3: Fast-clock method.

— **Slow-clock method**. In the generator in Fig. 4.4, the pulses from the Geiger detector in a given time window (i.e. the pulse from the clock is much larger than the rate of detection $\nu > \lambda$) increase the value of a counter: at this point there has been enough time to have registered many counts. When the clock produces a new pulse, the value of the counter is registered and the count resets. The output corresponds to the number of particle counts in each clock period.

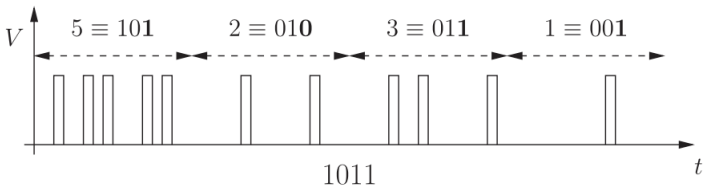


Figure 4.4: Slow-clock method.

— **Difference in time**. The time between two consecutive pulses is stored as t_1 and the same goes for the time between the next two pulses t_2. The random bits come from comparing the times. If $t_1 > t_2$ we output a 0 bit and if $t_1 < t_2$ we output a 1. Fig. 4.5 gives a graphical description of the method.

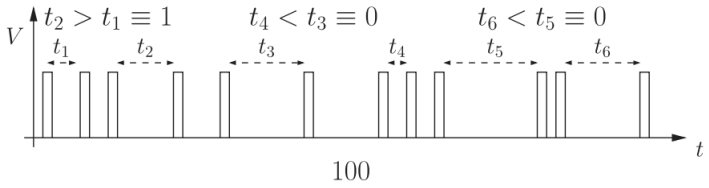


Figure 4.5: Difference in time.

A hybrid implementation of these methods has been developed in [Stanco *et al.*(2020)Stanco, Marangon, Vallone, Burri, Charbon, and Villoresi] (here at DEI in Padua): the great advantage in this case lies in avoiding the *waiting time* between each bit generation. This is an optimal algorithm in the sense that is able to extract "all available randomness".
Each time a photon hits a fast clock on the high pulse a 1 is registered. This doesn't result in a perfectly balanced $p(0)/p(1) = 50/50$ generator, but we can use some form of **debiasing** to clean up the sequence. Basically, we need a deterministic function - an **extractor**

$$\text{Ext} : \{0,1\}^m \to \{0,1\}^n \tag{4.15}$$

that take input strings of n bits $\{0,1\}^n$ into m output bits.
One possible method is the **Von Neumann extractor**: if $p(0) \neq p(1)$ we know that for sure the probability of the pairs $p(01) = p(10) = p(0) \cdot p(1)$. In this way the probability becomes balanced and from each pair we can extract one bit following the rule $01 \to 0$, $10 \to 1$ (the contrary is of course equally valid).
A better alternative is given by the **Peres extractor**, an extractor that does not throw away the equal-bit pairs 00 and 11. Two new sequences u and v are created:

— $\boldsymbol{u}$ is obtained by XOR-ing all bit couples, i.e. $00, 11 \to 0$ and $01, 10 \to 1$;

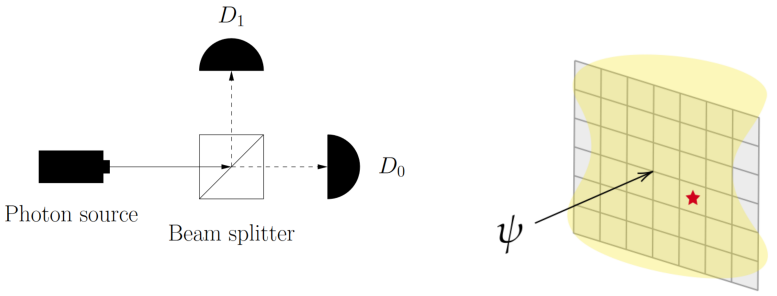
— $\boldsymbol{v}$ is obtained by considering only the second bit for equal-bit couples, i.e. $00 \to 0$ and $11 \to 1$.

The two sequences are now heavily biased: however, one can apply iteratively Von Neumann or Peres extractor once again to extract unbiased sequences. It can be demonstrated that the entropy of this procedure tends, for sufficiently long sequences to the Shannon entropy of the original sequence multiplied by its length. In this sense this is an example for an optimal algorithm to extract randomness.

Branching path generators

More sophisticated techniques involve (polarizing) beam splitters (PBS): referring to Fig. 4.6(a), by measuring the superposition of the two states $\frac{|1\rangle_1|0\rangle_2+|0\rangle_1|1\rangle_2}{\sqrt{2}}$ (i.e. a photon in the first and second path respectively) with a detector at the end of each

path will result in a click in just one of the detectors with $p = 1/2$ for each path. A generalization of this method consists in measuring the arrival position on one array of detectors of a spread out wave function (e.g. for an electron) as shown in Fig. 4.6(b).



(a) Balanced beam splitter schema.

(b) Light wave function over an array grid.

Figure 4.6: Polarizing beam splitter for two different configurations.

3.2 Trusted QRNGs: continuous variables

We focus now on the generation of RN in the case of continuous variables (i.e. infinite dimension Hilbert space). For example, for an EM radiation fixing a single mode is equivalent to fix the frequency ν, the propagation direction and the polarization. The only degree of freedom left is the number of photons that can increase indefinitely. A state could be written as one of the following

$$|0\rangle, |1\rangle, \ldots, |n\rangle$$

where $|0\rangle$ is the **vacuum state** and the indices inside each *ket* are in this case not just a label but assume a precise physical meaning (the number of photons).
We will now give some important definitions.

Definition 3.1 (Creation and annihilation operators)**.** An annihilation operator (usually denoted $\hat{a}$) lowers the number of particles in a given state by one. A creation operator (usually denoted $\hat{a}^\dagger$) increases the number of particles in a given state by one, and it is the adjoint of the annihilation operator.

$$\begin{aligned} \hat{a}^\dagger |n\rangle &= \sqrt{n+1}\,|n+1\rangle \\ \hat{a}\,|n\rangle &= \sqrt{n}\,|n-1\rangle \end{aligned} \qquad \left[\hat{a}, \hat{a}^\dagger\right] = 1 \tag{4.16}$$

These operators do not correspond to any observable since they are not hermitian, therefore they cannot be measured. By definition $\hat{a}\,|0\rangle$, i.e. *destroying the void*, generates a non-physical state.

Definition 3.2 (Number operator)**.** For systems where the total number of particles may not be preserved, the number operator is the observable that counts the number of particles. It is defined as

$$N = \hat{a}^\dagger \hat{a} \qquad\qquad N\,|n\rangle = n\,|n\rangle \tag{4.17}$$

For the current case it is related to the (quantum) intensity of the EM field since it counts the number of incoming photons.

Definition 3.3 (Coherent state)**.** In quantum optics the coherent state refers to a state of the quantized electromagnetic field that describes a maximal kind of coherence and a classical kind of behavior. It can be seen as a state in a system for which the ground-state wavepacket is displaced from the origin of the system. Mathematically, a coherent state $|\alpha\rangle$ is defined to be the (unique) eigenstate of the annihilation operator $\hat{a}$ with corresponding eigenvalue $\alpha \in \mathbb{C}$. Formally, this reads

$$\hat{a}\,|\alpha\rangle = \alpha\,|\alpha\rangle\,. \tag{4.18}$$

The formal solution of the eigenvalue equation is the vacuum state displaced to a location α in phase space

$$|\alpha\rangle = e^{-\frac{|\alpha|}{2}} \cdot \sum_0^\infty \frac{\alpha^n}{\sqrt{n!}}\,|n\rangle\,. \tag{4.19}$$

The probability of detecting n photons follows a Poisson distribution,

$$P(n) = |\langle n|\alpha\rangle|^2 = \frac{e^{-|\alpha|^2}|\alpha|^{2n}}{n!} \tag{4.20}$$

and the average photon number in a coherent state is $\langle n\rangle = \langle \hat{a}^\dagger \hat{a}\rangle = |\alpha|^2$. A classical EM wave can be described in the quantum realm by a coherent state. Moreover, the phase of $|\alpha\rangle$ is related to the phase of the oscillator.

The EM field can be **thought as an harmonic oscillator** and analogously to the classical case one could define position and momentum: these operators, written below as function of the creation/annihilation operators, are related to the quantum electric and magnetic field as shown

$$\hat{X} = \frac{\hat{a} + \hat{a}^\dagger}{\sqrt{2}} \to \hat{E} \qquad\qquad \hat{P} = \frac{\hat{a}^\dagger - \hat{a}}{\sqrt{2}} \to \hat{B} \tag{4.21}$$

If we denote the unitary eigenvector of the position operator corresponding to the eigenvalue x, then $|x\rangle$ represents the state of the particle in which we know with

certainty to find the particle itself at position x; with a similar formalism for the momentum we can write for every real position (momentum) x (p)

$$\hat{X}\,|x\rangle = x\,|x\rangle\,,\ x \in \mathbb{R} \qquad\qquad \hat{P}\,|p\rangle = p\,|p\rangle\,,\ p \in \mathbb{R}. \tag{4.22}$$

p and x are related through a Fourier transform

$$\langle x|\psi\rangle = \psi(x) \qquad\qquad \langle p|\psi\rangle = \tilde{\psi}(p) \tag{4.23}$$

where $\mathcal{F}[\psi(x)](p) = \tilde{\psi}(p)$. Now, *how can we use this formalism of the operators $\hat{X}$ and $\hat{P}$ to measure an optical field and extract RN?*

Homodyne measurement: introduction

Let's suppose to have a quantum state $\hat{\rho}_A$ on a single mode.
The homodyne measurement offers a simple way to measure the $\hat{X}$ operator (in a more general form, as we will see later). The experimental apparatus (Fig. 4.7) is made of a 50/50 beam splitter, on which we make our electromagnetic field entering in $\hat{a}$ interfere with a **local oscillator** $|\alpha\rangle_{\text{LO}}$ (i.e. a strong "classical" field **in the same mode** apart from the spatial direction) coming from $\hat{b}$. Along the direction $\hat{c}$ and $\hat{d}$ we place two photodiodes, that transform the electric field into an electrical current with which we measure the intensity.

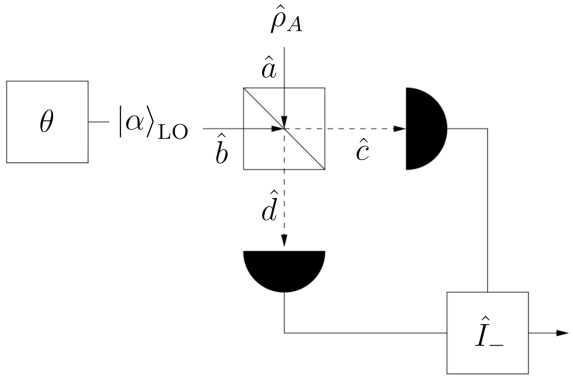


Figure 4.7: Homodyne measurement schema.

The quantity we measure $\hat{I}_-$ is the difference of these two currents, that will be clearly proportional to the difference in the number of incoming photons in $\hat{c}$ and $\hat{d}$. We want to express this quantity as a function of the input fields: then by measuring $\hat{I}_-$ the apparatus will retrieve a **tomography** (i.e. a complete description) of the

input state.

$$\hat{I}_- \propto N_d - N_c = \hat{d}^\dagger \hat{d} - \hat{c}^\dagger \hat{c} \tag{4.24}$$

$$\underset{(a)}{=} \frac{1}{2}\left[(i\hat{a}+\hat{b})^\dagger(i\hat{a}+\hat{b}) - (\hat{a}+i\hat{b})^\dagger(\hat{a}+i\hat{b})\right] = i(\hat{a}^\dagger\hat{b} + \hat{b}^\dagger\hat{a}) \tag{4.25}$$

$$\underset{(b)}{=} i(\hat{a}^\dagger\alpha + \hat{a}\alpha^\star) \underset{\alpha=|\alpha|e^{i\theta}}{=} i|\alpha|(\hat{a}^\dagger e^{i\theta} + \hat{a}e^{-i\theta}) \underset{(c)}{=} i\sqrt{2}|\alpha|\hat{X}_\theta \tag{4.26}$$

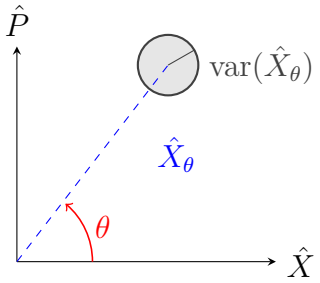
where in (a) we remembered that a 50/50 beam splitter operator [5] acts on the input modes as

$$\mathrm{BS} = \begin{pmatrix} 1 & i \\ i & 1 \end{pmatrix} \qquad \begin{pmatrix} \hat{c} \\ \hat{d} \end{pmatrix} = \mathrm{BS} \begin{pmatrix} \hat{a} \\ \hat{b} \end{pmatrix}. \tag{4.27}$$

In (b) we applied Eq. (4.18) whereas in (c) we defined the **quadrature observable** $\hat{X}_\theta$ as

$$\hat{X}_\theta \equiv \frac{\hat{a}e^{-i\theta} + \hat{a}^\dagger e^{i\theta}}{\sqrt{2}}. \tag{4.28}$$

We change the phase θ by varying the one of the local oscillator $|\alpha\rangle_{\mathrm{LO}}$: in this way we can vary also the value of $\hat{X}_\theta$.
$\hat{X}_\theta$ can be represented in the bi-dimensional phase space $\hat{X}, \hat{P}$ (below) as a straight line; by varying θ, i.e. rotating $\hat{X}_\theta$ we can span the whole plane (practically this is done by sampling) and consequently retrieve the physical properties of the field, i.e. our quantum state. This corresponds to a quantum tomography of $\hat{\rho}$.



The most important functions that define the quantum state and that can be reconstructed from $\hat{X}_\theta$ are

$$Q(\alpha) = \frac{1}{\pi}\langle\alpha|\,\hat{\rho}\,|\alpha\rangle \qquad \text{(Husimi)}$$

$$\hat{\rho} = \int P(\alpha)\,\langle\alpha|\alpha\rangle\,d^2\alpha \qquad \text{(P-function)}$$

[5] the general beam-splitter operator, setting t, r to be respectively the transmission and reflection coefficients that have to respect the constraint $|t+r|^2 = 1$, has the form $\mathrm{BS} = \begin{pmatrix} t & r \\ r & t \end{pmatrix}$

The former, called the **Husimi function** is a distribution commonly used in quantum mechanics to represent the **phase space distribution of a quantum state** such as light and represent the expectation value of the coherent state $|\alpha\rangle$ over $\hat{\rho}$. The latter $\hat{\rho}$ is expressed as an incoherent superposition of coherent states. Moreover, the $P(\alpha)$ in (P-function) could be in principle negative. However, if $P(\alpha)$ is a positive function, then the state $\hat{\rho}$ would be the usual classical mixture state. Another important observable is given by

$$W(\alpha) = \frac{1}{\pi^2}\int \exp(\eta^\star\alpha - \eta\alpha^\star)\chi(\eta)d^2\eta \qquad \text{(Wigner function)}$$

$$\chi(\eta) = \mathrm{Tr}\left[\hat{\rho}\exp\left(\eta^\star\hat{a}^\dagger - \eta\hat{a}^\star\right)\right] \qquad \text{(Characteristic function)}$$

in which $\eta \in \mathbb{C}$. This can be seen as a anti-Fourier transform of the characteristic function $\chi(\eta)$. From this definitions we can rewrite $Q(\alpha)$, $P(\alpha)$ as

$$Q(\alpha) = \frac{1}{\pi^2}\int \exp(\eta^\star\alpha - \eta\alpha^\star)\chi(\eta)\exp\left(-\frac{1}{2}|\eta|^2\right)d^2\eta \tag{4.29}$$

$$P(\alpha) = \frac{1}{\pi^2}\int \exp(\eta^\star\alpha - \eta\alpha^\star)\chi(\eta)\exp\left(+\frac{1}{2}|\eta|^2\right)d^2\eta \tag{4.30}$$

So, all in all, $W(\alpha)$, $Q(\alpha)$, $P(\alpha)$ are different representation of the same quantity: usually the Wigner representation is preferred due to its regularity property.
From this expression we can also retrieve the physical intuition behind the formulas, in particular for $Q(\alpha)$. The latter can be seen as the anti-Fourier transform of the product of $\mathcal{F}(\chi)[\alpha] \equiv W(\alpha)$ and the F. transform of a gaussian wave-packet (which is again a gaussian); since the product of transforms is equivalent to the transform of a convolution, Eq. (4.30) is the mixture of a state $\hat{\rho}$ and the vacuum ($\sim \mathcal{N}\left(\alpha, \frac{1}{2}\right)$) as we will see in the following section.

It is important to stress that $W(\alpha)$ and $P(\alpha)$ are not probability distribution: they are **quasi-probability distributions**: they satisfy the normalization constraint but they can assume negative values. On the other hand, $Q(\alpha)$ is indeed a probability distribution.
The probability distribution of $\hat{X}_\theta$ is obtained by integrating out $W(\alpha)$ in its orthogonal direction: $\alpha = x + ip$, $x, p \in \mathbb{R}$ then $W(\alpha) \equiv W(x, p)$. To retrieve the probability distribution of the x variable $f(x)$ we can just integrate over p:

$$f(x) = \int W(x, p)dp.$$

The Homodyne plays a central role both in QKD and in QRNG. *How can we now extract RN?*

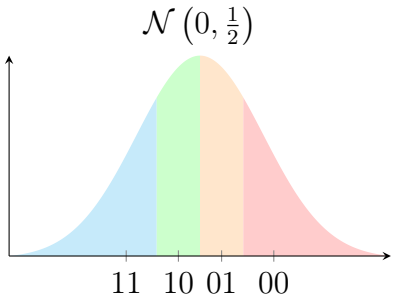
Vacuum measurement

The framework is the Homodyne measurement presented above.
If we now consider as input the void state $|0\rangle$, that still is perturbed by **quantum noise**; in fact by performing the calculations,

$$\langle x|0\rangle = \psi_0(x) = \frac{1}{\pi^{\frac{1}{4}}} \exp\left(-\frac{x^2}{2}\right) \tag{4.31}$$

$$|\psi_0(x)|^2 = \frac{1}{\sqrt{\pi}} \exp\left(-x^2\right) \tag{4.32}$$

we find the wave function of the vacuum is a Gaussian distribution $\mathcal{N}\left(\alpha, \frac{1}{2}\right)$. Then, by measuring the $\hat{X}$ operator I will retrieve a random variable $x \sim \mathcal{N}\left(0, \frac{1}{2}\right)$.
So this quantum intrinsic fluctuation, that is not due to any thermal noise, becomes negligible (in particular in the case of high intensity field) in the classical limit and will be the source of the random bits. To obtain a uniform distribution from a gaussian we will associate each one of the 2^n equal-area quantile, depending on the length n of the bits string we want to obtain, to a particular number.



When computing the entity of this fluctuation ΔX^2 what we get is

$$\Delta X^2 = \langle \hat{X}^2 \rangle - \langle \hat{X} \rangle^2 = \frac{1}{2} \neq 0, \forall\, |\alpha\rangle\,, \tag{4.33}$$

i.e. there is an intrinsic fluctuation of the field. If we replace the vacuum with a coherent state we get another gaussian distribution for $\hat{X}$, this time centered around the value of α (and not 0); still, as stated in Eq. (4.33) the relation for the variance holds for all coherent states.

Random phase noise of laser

The output of a laser has a random phase of quantum origin that can be used to produce random bits. Laser is, keeping it low-key, the amplification of a spontaneous emission process. In short, it amplifies the void-noise to a coherent phase

$$|0\rangle \underset{\text{LASER}}{\longrightarrow} |\alpha\rangle\,. \tag{4.34}$$

Since the global phase of the vacuum state $|0\rangle$ is not observable, the phase that will be assigned to $\alpha = |\alpha|e^{i\theta}$ is completely random. So, at every pulse i, that is determined by the lasing threshold, we get a θ_i random phase. In this way we are encoding the quantum noise (i.e. the random information) into a classical signal: in fact the **full device and measurements are completely classical**. The only quantum contribution is the "vacuum amplification" at the beginning, i.e. the lasing process.

Despite measuring the absolute phase of a state is not possible, the phase differ-

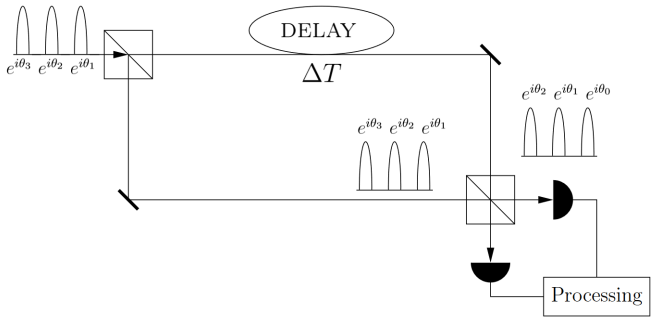


Figure 4.8: Random phase noise schema: pulses are ΔT apart.

ences of couples of pulses can be obtained. The devices used is basically a beam splitter, to which a delay line is added on a single branch (Fig. 4.8): this introduces a delay equal to the period ΔT between two pulses. What we are building is an **interferometer between two pulses**: this will highlight their phase differences via an intensity pattern. Indeed for two different pulses with phases θ_1, θ_2 the intensity will be proportional to $I \propto \cos^2(\theta_1 - \theta_2)$. Randomness is then retrieved by this phase-difference measure.
The same can be reached with a continuous laser: if the introduced delay is far above the coherence time (the time for which the laser maintains its phase) of the laser $\Delta T \gg \tau_{coh}$, the phase difference $\Delta\phi(t)$ is again a Gaussian random variable of a mean that tends to 0.

These methods are very useful because, for once, we use standard communication components. Moreover, these devices can reach high generation rate: single photon detector ($\sim 100\,\text{MHz}$ of operations) suffer from the problem of dead time, i.e. the detector becomes again usable after a latency time of $\sim 100\,\text{ns}$, which corresponds to tens of MHz of operations. However, photodiodes can up go to tens of GHz and are thus extremely faster and can improve significantly the generation rate.

3.3 Semi-device independent QRNGs

Up to now we have dealt in a special scenario: a pure state undergoes a measurement in a fully controlled environment to yield to a classical outcome. One can then compute the probability of the latter as $P_2 = \mathrm{Tr}\left(\hat{\rho}_A \hat{F}_2\right)$ and consequently obtain the usual measures of entropy. However, this situation is far from reality: for example, just producing pure states in a laboratory is practically impossible.
Following these considerations we wold like now to relax the hypotheses in the cases where the source is not fully characterized. As a first example we deal with the **worst-case scenario** in which the whole source is in the hands of an eavesdropper.

Source-device independence: projective qubit measurement

The first idea comes from [Fiorentino *et al.*(2007)Fiorentino, Santori, Spillane, Beausoleil, and Munro]. The situation is that of an unknown quantum state $\hat{\rho}_A$ fully controlled by Eve and a usual POVM $\{\hat{F}_x\}$. *What is now the guessing probability?*

It can be proven that

$$P_g(X|E) = \max_{\{P_n, \hat{\tau}_n\}} \underbrace{\sum_n P_n \underbrace{\max_{\{x\}} \left[\mathrm{Tr}\left(\hat{F}_x \hat{\tau}_n\right)\right]}_{\text{most probable outcome}}}_{\text{average guessing probability}} . \tag{4.35}$$

Since any density matrix (any quantum state in fact) can be decomposed as

$$\hat{\rho}_A = \sum_n p_n \hat{\tau}_n = \sum_n p_n \left|\psi_n\right\rangle \left\langle\psi_n\right| \tag{4.36}$$

given a collection of pure states $|\psi_n\rangle$, Eve has basically control over which base $\hat{\tau}_n$ to choose. Eve cannot control the outcome of a measurement on the pure state $|\psi_n\rangle$ (because these probabilities are governed solely by the laws of QM), but knows at each time the state Alice is measuring. *How much information can Eve obtain in this case about Alice's random sequence?*
At this point, Eve can maximize over all possible decomposition the argument of Eq. (4.35). For any state that Eve sends she can guess the most probable outcome and thus adopt the optimal strategy for a given possible state $\hat{\rho}_A$.
For the very simple scenario of a projective qubit measurement we assume that we can measure the state in all Pauli matrices $\hat{\sigma}_{x,y,z}$: this is equivalent to perform a quantum tomography. We thus have full knowledge of $\hat{\rho}_A$. In this situation $\hat{\rho}_A$ is known but its decomposition $\sum_n P_n \hat{\tau}_n$ is unknown and in full control of Eve as mentioned above. If $\hat{\rho}_A$ is a pure state, then there is only one term in the

decomposition of Eq. (4.36) and the P_g can be explicitly calculated (since there is no sum nor maximization),

$$P_g = \max\left\{|\langle 0|\psi\rangle|^2,\, |\langle 1|\psi\rangle|^2\right\} \tag{4.37}$$

and thus (for a single random bit generated, $n = 1$)

$$H_{\min}(|\psi\rangle\langle\psi|) = -\log_2 P_g, \tag{4.38}$$

which coincides in this case to the classical min-entropy. Again, in general $\hat{\rho}_A$ is not pure, and Eve will have the control over its decomposition. So, we extended this discussion to a decomposition such as the one on the RHS of Eq. (4.36) (and for a generic number n of generated random bits),

$$H_{\min}\left[\left(\sum_n p_n |\psi_n\rangle\langle\psi_n|\right)^{n}\right] = -n\sum_n p_n \log_2 P_g = n\sum_n p_n H_{\min}\left(|\psi_n\rangle\langle\psi_n|\right) \tag{4.39}$$

Since $\hat{\rho} = \frac{1}{2}(\mathbb{1} + \vec{\sigma}\cdot\vec{r})$, $\|\vec{r}\| = 1$ for pure states, we can calculate the two output probabilities along the axis z (by measuring the operator σ_z) as

$$P(0) = \frac{1+r_3}{2} \qquad\qquad P(1) = \frac{1-r_3}{2}.$$

Since the state is pure, this can be represented by a vector pointing **on the surface** of the Bloch sphere with the condition $r_1^2 + r_2^2 + r_3^2 = 1$: solving for r_3 yields $r_3 = \pm\sqrt{1-r_1^2-r_2^2}$. The maximum is given, independently of the sign, by $\frac{1}{2}(1 + \sqrt{1-r_1^2-r_2^2})$. We define

$$f(|\psi\rangle\langle\psi|) \equiv -\log_2\left(\frac{1+\sqrt{1-r_1^2-r_2^2}}{2}\right) = H_{\min}(|\psi\rangle\langle\psi|). \tag{4.40}$$

More in general, the minimum value $\tilde{H}_{\min}$ of the min-entropy of a system taken over all possible decompositions of $\hat{\rho}_A$ is given by

$$\tilde{H}_{\min}(\hat{\rho}_A) = f(\hat{\rho}_A), \tag{4.41}$$

where is important to notice that the system is described now by an **arbitrary** density matrix $\hat{\rho}_A$.

We want now to prove this result. We already showed the equivalence for pure states in Eq. (4.40). Moreover, $f(\hat{\rho}_A)$ is a convex function of $\vec{r}$ on the Bloch's sphere. Using the convexity of f we can write[6]

$$f(\hat{\rho}_A) = f\left(\sum_n p_n |\psi_n\rangle\langle\psi_n|\right) \leq \sum_n p_n f(|\psi_n\rangle\langle\psi_n|) \tag{4.42}$$

[6]Let X be a convex subset of a real vector space and let $f : X \to \mathbb{R}$ be a function. Then f is called convex $\iff \forall\, 0 \leq t \leq 1, \forall x_1, x_2 \in X$ it holds that $f(tx_1 + (1-t)x_2) \leq tf(x_1) + (1-t)f(x_2)$.

for each decomposition of $\hat{\rho}_A$. Using Eq. (4.39) we obtain

$$f(\hat{\rho}_A) \leq H_{\min}\left(\sum_n p_n \left|\psi_n\right\rangle \left\langle\psi_n\right|\right) \tag{4.43}$$

indicating that $f(\hat{\rho}_A)$ is a lower bound for $\tilde{H}_{\min}(\hat{\rho}_A)$. When $\hat{\rho}_A$ is a pure state we proved in Eq. (4.40) that the equivalence holds: therefore $f(\hat{\rho}_A)$ is equal to the minimum of $H_{\min}$ over all possible decomposition of $\hat{\rho}_A$, i.e. the claim in Eq. (4.41).

For a maximally mixed state $\hat{\rho}_A = \mathbb{1}_A$, $\vec{r} = 0$ and consequently $P_g = 1$, i.e. the adversary has full knowledge of the system. Indeed for this case $H_{\min} = 0$ and no randomness can be extracted. For the other cases a lower bound on the extractable randomness can be found through this measure.
This gives us a hint also on the best strategy Eve could adopt, namely send pure states (as anticipated above) that have the same projection on the $z = 0$ plane as $\hat{\rho}_A$. Formally, given $\vec{r}_{\hat{\rho}_A} = \left(r_1, r_2, r_3\right)$, $\vec{r}^{\,\text{best}}_{E_{1,2}} = \left(r_1, r_2, \sqrt{1 - r_1^2 - r_2^2}\right)$ E tries to maximize the $\hat{z}$ component, constrained on the values of r_1 and r_2, which are fixed by the choice of $\hat{\rho}_A$ (to be sent).
Eve, following her own interests, could freely decide to send $|0\rangle$ and $|1\rangle$ (states with the largest $\hat{z}$ component, corresponding to $r_1 = r_2 = 0$): now, each state that Alice measures would be instantly known to Eve. By sending this states Eve would guess correctly the state measured by Alice with a probability 1 (substitute the values for $r_{1,2}$ in Eq. (4.40)). In this case we do not trust the source and no randomness can be extracted from it (at least for cryptographic purposes).
On the other hand, from the point of view of Alice, the best states she can receive are those on the equator, $\frac{1}{2}\left(|0\rangle + e^{i\theta}|1\rangle\right)$, i.e. the so called **maximally entangled states**: this implies $P_g = \frac{1}{2}$ (random guess) and thus $H_{\min} = 1$. For all the states in between a measure of trust for the source is given indifferently by $\frac{1}{2} < P_g < 1$, $0 < H_{\min} < 1$.

Source-device independence: general case

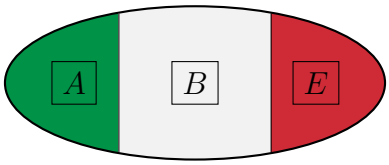
The results obtained in the previous section can be generalized to the more complex scenarios not limited to projective qubit measurements. Indeed, it can be shown that we don't need a full tomography of the state $\hat{\rho}_A$ but it is sufficient to know its **purity** and for this scope the state can be measured in two different basis and not on the whole set of Pauli matrices as before. We remember that differences between classical and quantum system arises when measuring in more than one basis.[7]
This idea allows to express **Heisenberg's uncertainty principle** (if I have two basis, I cannot measure with arbitrary precision in both of them at the same time)

[7]For example, the states $|+\rangle\langle+|$ and $\mathbb{1}/2$ are identical when measured in the $\hat{z}$ basis but have completely different outcomes when measured in the $|+\rangle$, $|-\rangle$ basis.

in terms of quantum mean-entropies; in [Tomamichel and Renner(2011)] the authors demonstrated that

$$H_{\min}(Z|E) + H_{\max}(X|B) \geq -\log_2 C \qquad C = \max_{x,z} \left\| \sqrt{\hat{M}_z}\sqrt{\hat{N}_x} \right\|_\infty^2 \tag{4.44}$$

where A, B and E are three systems and the measurement (in the most general case POVM) are chosen either from the set $\{\hat{M}_z\}$ or from $\{\hat{N}_x\}$. This shows that the sum of the two mean-entropies **is lower bounded** and delivers useful information without explicitly computing the exact values for both $H_{\min}$, $H_{\max}$.



Again the situation after the measurement will be analogous to what has been seen in Eq. (4.12) for a classical quantum density matrix,

$$\hat{\rho}_{ABE} \underset{\text{MEASURE}}{\longrightarrow} \sum_z P_z \left|z\right\rangle \left\langle z\right| \otimes \hat{\rho}^z_{BE} \tag{4.45}$$

The C factor in Eq. (4.44) represents the **overlap** between the two measurements. Let's consider the special case of projective measurements. In this case, C can be rewritten in the form

$$C = \max_{x,z} \left|\langle z|x\rangle\right|^2; \tag{4.46}$$

if $\{\hat{M}_z\}$, $\{\hat{N}_x\}$ share a common projector the bound becomes trivial ($-\log_2 C = 0$ since the max eigenvalue for a projector is $1 = C$). Moreover, C has a minimum for **mutually unbiased basis** (MUB) when it holds $C = \frac{1}{d}$.
The result in Eq. (4.44) is general: for the special case where B is trivial (i.e. is not present or in contact with the other two) we obtain

$$H_{\min}(Z|E) + H_{\max}(X) \geq -\log_2 C \underset{(a)}{\Longrightarrow} \tag{4.47}$$

$$H_{\min}(Z|E) \geq \log_2 d - 2\log_2 \sum_x \sqrt{p(x)} \tag{4.48}$$

where in (a) we considered MUB and used the relation seen in Eq. (4.9) for the separable case, i.e. $H_{\max}(X) = 2\log_2 \mathrm{Tr}\left(\sqrt{\hat{\rho}_X}\right) \equiv H_{1/2}(X)$. This means that by **measuring a system in one base we can bound the mean entropy on another base**. This is evident in the case of a simple system composed of one qubit,

$$|+\rangle \qquad p_+(x) = 1, p_-(x) = 0 \longrightarrow H_{\min} \geq 1 \tag{4.49}$$

$$\mathbb{1}/2 \qquad p_+(x) = p_-(x) = 1/2 \longrightarrow H_{\min} \geq 0. \tag{4.50}$$

In the first example we have at least one random bit generated per measurement, while the second is useless for it, as it is bounded by 0 (because this state can be correlated with the environment). Eq. (4.47) is a generalization of the guessing probability P_g in Eq. (4.40) and can be applied also in the case of > 2-dimensional basis. Moreover, this approach requires measurements only in two basis and not a full tomography as anticipated previously.
With a probability $p_{\text{switch}} < p_{\text{gen}}$ some bits are used to switch between the two basis. p_{gen} is the probability for a bit to be used to generate RN and is greater in order for the algorithm to be efficient.

To be precise both this last case and [Fiorentino *et al.*(2007)Fiorentino, Santori, Spillane, Beausoleil, and Munro] are not RNG in the strict sense but rather **randomness expander**: they require a **true initial random seed** with which they can initiate the sequence. The algorithm is then self-sustained, since part of the bits are used to perform true random choices in the RN generation process.

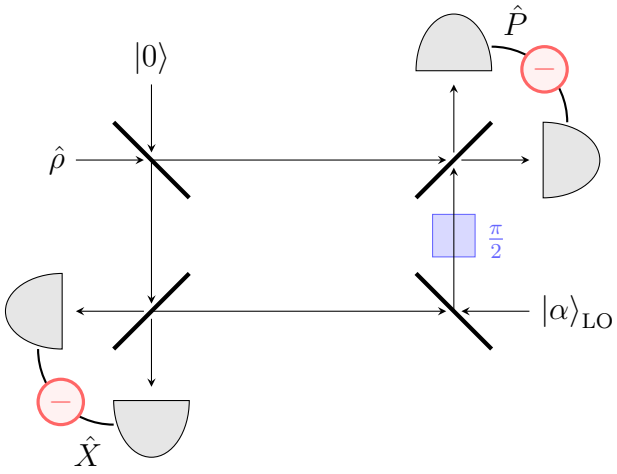
This results were also generalized to the case of **continuous variables**; we mention only the result

$$H_{\min}(\hat{X}|E) + H_{\max}(\hat{P}|E) \geq -\log_2 C(\delta_x, \delta_p) \qquad C = \frac{\delta_x \delta_p}{2\pi} S_0\left(1, \frac{\delta_x \delta_p}{4}\right)^2 \tag{4.51}$$

where S_0 is a spheroidal function, a function of the resolution of the measurement. So also the x, p quadrature is subjective to the uncertainty principle. We will not go further into details: it is sufficient to know that in this case the generation rate can be $\geq 1\,\text{Gbit}\,\text{s}^{-1}$ (using photodiodes can be much more efficient, as already seen in the last paragraph of Section 3.2).

Heterodyne measurement

We conclude this section with an example of heterodyne measurement, i.e. a device that produces interference between the injected field $\hat{\rho}$ and the local oscillator, with the difference with respect to the homodyne measurement that now the two have different frequencies that interfere. The device in the case of a **double homodyne** is schematized below.



The H. uncertainty principle is not violated in this case because we are not measuring the actual state $\hat{\rho}$ but rather a perturbed version (its convolution) with a vacuum mode $|0\rangle$. It can be demonstrated that **this measure is actually performed on the coherent state**, i.e. a POVM given by

$$\hat{\Pi}_\alpha = \frac{1}{\pi} |\alpha\rangle \langle\alpha| \qquad\qquad \alpha = x + ip. \tag{4.52}$$

Note that $\hat{\Pi}_\alpha$ is not a projector because of the pre-factor $\frac{1}{\pi}$. Calculating the probability

$$P_\alpha = \mathrm{Tr}\left(\hat{\rho}\hat{\Pi}_\alpha\right) = \frac{1}{\pi} \langle\alpha|\hat{\rho}|\alpha\rangle \equiv Q(\alpha) \tag{4.53}$$

we obtain the **Husimi function**. This apparatus allows us to directly measure the Husimi function.

In Eq. (4.35) we saw that

$$P_g(X|E) = \max_{\{P_n, \hat{\tau}_n\}} \sum_n P_n \underbrace{\max_{\{x\}} \left[\mathrm{Tr}\left(\hat{F}_x \hat{\tau}_n\right)\right]}_{(*)}; \tag{4.54}$$

For a generic class of measurements one can upper-bound the argument $(*)$ by simply maximizing it over all possible choices of $\hat{\tau}$, obtaining the maximum probability outcome for any POVM $P^\star$ **independently of the input state**,

$$\max_{\{x\}} \mathrm{Tr}\left(\hat{F}_x \hat{\tau}_k\right) \leq \max_{\{x,\hat{\tau}\}} \mathrm{Tr}\left(\hat{F}_x \hat{\tau}\right) \equiv P^\star \implies P_g(X|E) \leq P^\star \tag{4.55}$$

since $\sum_n P_n = 1$. For a pure state the value is trivially $P^\star = 1$; however, for a POVM the guessing probability cannot reach 1 independently of the injected state. In the heterodyne case

$$Q(\alpha) = \frac{1}{\pi} \langle\alpha|\rho|\alpha\rangle \leq \frac{1}{\pi} \quad \text{never 1!} \tag{4.56}$$

i.e whenever we perform a measurement in the heterodyne scheme we have an upper-bound in the output probabilities that implies and **upper-bound on the guessing probabilities**.[8]
The advantage of this technique is that it does not require any random seed, since it is carried out by a fixed measurement. Following this idea, we could see it as if the POVM had some intrinsic quantum randomness independently of the state that can be exploited.

Measurement-device indipendence: projective qubit measurement

The scenario now is that of a trusted source but untrusted measurement. The idea is always that of a single measurement device: this time we can change the input state in order to characterize the measurement and determine the randomness one can extract from a single measurement state.
In [Cao *et al.*(2015)Cao, Zhou, and Ma] the authors show that by adopting this procedure, in a very similar fashion to a quantum tomography (this time of a measurement and not of a state, i.e. a **process tomography**), one could compute the coefficients of

$$\hat{\Pi}_0 = a_0(\mathbb{1} + \vec{n} \cdot \vec{\sigma}) \tag{4.57}$$

while for the mean entropy it holds that

$$H_{\min} = 2a_0 H_\infty \left(\frac{1 + \sqrt{1 - n_y^2 - n_z^2}}{2} \right) \qquad H_\infty(p) = -\log_2 \max(p, 1-p). \tag{4.58}$$

Dimension witness

We can relax the conditions even further and consider both source and measurement as black boxes.

In [Lunghi *et al.*(2015)Lunghi, Brask, Lim, Lavigne, Bowles, Martin, Zbinden, and Brunner] the authors propose a **the self-testing QRNG protocol** (Fig. 4.9, again it is actually a randomness expander). The protocol uses two devices which respectively prepare and measure an uncharacterized qubit system. In each round of the protocol, the observer chooses settings among four possible preparations, $x = 0, 1, 2, 3$, and two measurements $y = 0, 1$, resulting in a binary outcome $b = \pm 1$. To model imperfections, the internal state of each device is represented by a random variable λ for the preparation device and μ for the measurement device - which are unknown to the observer. The devices are assumed independent (scenario where the devices are not maliciously conspiring against the user), i.e. $p(\lambda, \mu) = q(\lambda)r(\mu)$

[8] Things are actually trickier: $Q(\alpha)$ and P_g are a probability density functions and not a probability distributions since we are considering a discretized set of outcomes and not a continuum space.

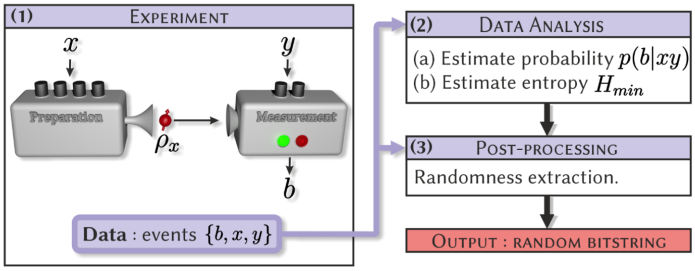


Figure 4.9: Sketch of the self-testing QRNG protocol.

where $\int q(\lambda)d\lambda = \int r(\mu)d\mu) = 1$. In each round of the experiment, the preparation device emits a qubit state ρ_x^λ which depends on the setting x and on the internal state λ. Similarly, the measurement device performs a measurement M_y^μ . Thus the distributions of λ and μ determine the distributions of the prepared states and the measurements. The task of the observer is to estimate the amount of genuine quantum randomness generated in this setup, based only on the observed distribution $p(b|x, y)$.

The protocol allows the observer to separate quantum randomness from the randomness due to technical noise. The key technical tool is a function, which works as a "dimension witness". Given data $p(b|x, y)$, the quantity

$$W = \det \begin{pmatrix} p(1|0,0) - p(1|1,0) & p(1|2,0) - p(1|3,0) \\ p(1|0,1) - p(1|1,1) & p(1|2,1) - p(1|3,1) \end{pmatrix} \tag{4.59}$$

captures the *quantumness* of the preparation and measurements. Specifically, if the preparations are classical (i.e. there exist a basis in which all states ρ_x^λ are diagonal), one has that $W = 0$, while a generic qubit strategy achieves $0 \leq W \leq 1$. $W > 0$ guarantees that the measurements performed are incompatible and since it is then impossible to simultaneously assign deterministic outcomes to them, this enables us to bound the guessing probability and certify randomness. Given x, y, and knowledge of the internal states λ, μ, the best guess for b is given by $\max_b p(b|x, y, \lambda, \mu)$. Assuming uniformly distributed x and y, the average probability of guessing b fulfills

$$P_g = \frac{1}{8} \sum_{x,y,\lambda,\mu} q_\lambda r_\mu \max_b p(b|x, y, \lambda, \mu) \leq \frac{1}{2} \left(1 + \sqrt{\frac{1 + \sqrt{1 - W^2}}{2}} \right) \tag{4.60}$$

and therefore the guessing probability can be upper-bounded by a function of W, which can be determined directly from the data $p(b|x, y)$. Finally, to extract random bits from the raw data, a randomness extraction procedure is used. The number of random bits that can be extracted per experimental run is given by the min-entropy $H_{\min} = -\log_2 P_g$.

Note that randomness can be extracted for any $W > 0$, since $P_g < 1$ in this case.

The maximal value of $W = 1$ can be reached using the set of preparations and measurements of the BB84 protocol ($x = |0\rangle, |1\rangle, |+\rangle, |-\rangle$, $y = \hat{\sigma}_z, \hat{\sigma}_x$). In this case, randomness can be certified with min-entropy $H_{\min} \simeq 0.2284$. Using other preparations and measurements, e.g. if the system is noisy or becomes misaligned, one will typically obtain $0 < W < 1$. Nevertheless, for any value $W > 0$, randomness can be certified, and the corresponding min-entropy can be estimated using Eq. (4.60).
The protocol is therefore self-testing, since the evaluation of W allows quantifying the amount of randomness in the data.

3.4 Full device indipendence

The framework is now that of Bell inequalities. The generator, first proposed in [Pironio *et al.*(2010)Pironio, Acín, Massar, de la Giroday, Matsukevich, and et al.], does not require any assumption on the internal working of the devices.
The apparatus consists in two devices A and B that accept as input $x = 0, 1$, $y = 0, 1$ respectively and return as output $a, b \in \{-1, 1\}$. This can be modelled by a simple 2-qubits system. If $\langle a_x b_y \rangle$ is the average value of the product of the outcomes a, b for a fixed choice of the measures x, y, explicitly

$$\langle a_x b_y \rangle = \sum_{a,b \in \{-1,1\}} ab \cdot p(a, b|x, y). \tag{4.61}$$

The idea now is to consider the variable S i.e. the **Bell parameter**, function of the correlations $\langle a_x b_y \rangle$ and defined as

$$S = \langle a_0 b_0 \rangle + \langle a_0 b_1 \rangle + \langle a_1 b_0 \rangle - \langle a_1 b_1 \rangle \underset{(a)}{\Longrightarrow} \tag{4.62}$$

$$S = \sum_{x,y} (-1)^{xy} \left[p(a = b|x, y) - p(a \neq b|x, y) \right] \tag{4.63}$$

where in (a) we used $\langle ab \rangle = p(a = b) - p(a \neq b)$. For any classical system it holds $S \leq 2$: in this context some correlation is necessarily pre-determined. If in contrast $2 < S < 2\sqrt{2}$ we violate Bell's inequality and we can certify that there is entanglement (i.e. *quantumness* in the source) without actually ever "opening" the black box of system and measurement. Any measurement contains therefore some form of intrinsic randomness due to the very nature of quantum theory.
In practice, the proposal was to upper-bound the probability $p(a|x)$ (the probability of the output given the input). For $2 < S < 2\sqrt{2}$ some minimum degree of entanglement is required, otherwise the random value cannot be trusted. Let's try to quantify it.
A generic 2-qubits entangled system can be always written as (Schmidt decomposition)

$$|\psi\rangle_\theta = \cos\theta \, |00\rangle + \sin\theta \, |11\rangle \tag{4.64}$$

with $0 < \theta \leq \frac{\pi}{4}$ in order to allow some degree of violation. The Bell parameter is upper bounded by θ ($2 < S < 2\sqrt{2}$) through

$$S_\theta \leq 2\sqrt{1+\sin^2 2\theta} \tag{4.65}$$

$$\left(\implies -\sin^2 2\theta \leq 1 - \frac{S_\theta^2}{4} \right). \tag{4.66}$$

This tells us that if we decrease the entanglement (lower θ) we get a lower violation. We can compute the maximum output probability focusing on the A system: this is given by the maximum eigenvalues of the reduced density matrix $\hat{\rho}_A = \underset{B}{\mathrm{Tr}} \left(|\psi\rangle_\theta \langle\psi|_\theta \right) = \cos^2 |0\rangle \langle 0| + \sin^2 |1\rangle \langle 1|$, i.e. $\cos^2\theta$. One can thus upper bound the probability

$$p(a|x) \leq \cos^2\theta = \frac{1+\cos 2\theta}{2} = \frac{1+\sqrt{1-\sin^2 2\theta}}{2} \underset{4.66}{\leq} \frac{1}{2}\left(1+\sqrt{2-\frac{S_\theta^2}{4}}\right) \tag{4.67}$$

thus relating $p(a|x)$ to the parameter S_θ. If there is no violation we get a useless bound, $S_\theta = 0 \implies p(a|x) \leq 1$; if, on the other hand, we violate the inequality, Eq. (4.67) tells us that the guessing probability cannot reach 1: again, there is some intrinsic randomness that we can exploit to our needs.

Chapter 5

Randomness extractors

When building QRNGs, the number extracted thanks to Nature are far from ideal, in the sense that they are **not independent** and **not uniformly distributed**. In general, given a sequence of bits, we can extract a shorter one that is instead perfectly balanced between 0 and 1. Here, extractors are typically used for **privacy amplification**, meaning to turn a partially secure raw key (about which the adversary may have non-trivial information) into a perfectly secure key (typically shorter). We thus demand that the extractor output is **uniform with respect to the side information** held by the eavesdropper or a more general adversary.

1 Extractors quality and side information

No side information

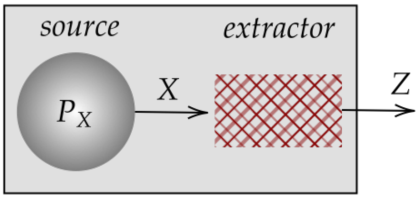


Figure 5.1: No side-information source extractor.

We can model our framework (Fig. 5.1) as a black box composed of a source (which outputs a random variable X and is characterized by a PDF P_X) and an extractor module, i.e. a mapping function $X \to Z$, which returns the final value Z that we have access to and that we wish to be completely random.
We have seen in Definition 2.19 that a security measure for a mechanism M can be expressed in terms indistinguishability from its ideal counterpart $M^\star$; having to deal with composite systems we wish to exploit the composability property of distinguishability (Proposition 2.2). In Corollary 2.1 we have seen that the variational

distance $d_V(\cdot,\cdot)$ can be exploited to measure distinguishability. For this reason, we shall refer to d_V as an index of randomness for the whole apparatus: the goal is to measure how far is the distribution of Z of our black box from the ideal one $Z^\star$, which is i.i.d. over a uniform distribution. For this fist simple case we have that:

$$d_V(Z, Z^\star) = \frac{1}{2}\sum_{z\in\mathcal{A}} |P_Z(z) - P_{Z^\star}(z)|, \tag{5.1}$$

where $\mathcal{A}_Z$ is the alphabet for z and $P_{Z^\star}(z) = \dfrac{1}{|\mathcal{A}_Z|}\,\forall z \in \mathcal{A}_Z$ (uniform distribution). Sometimes the computation of d_V can be expensive, hence it is helpful (and sometimes sufficient) to give an upper bound to it by exploiting the triangular inequality.

With classical side information

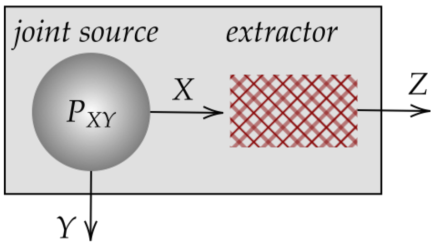


Figure 5.2: Side-information source extractor.

A more interesting case is when we have side information, meaning that our source will not direct all its generations towards the extractor but allows some quantity of information to escape from the side. We can model it stating that the source draws couples (X, Y) from a joint probability distribution P_{XY}. In a real case scenario, the variable Y corresponds to an information leakage and it is what the eavesdropper can exploit to know Z, since Y comes from a joint distribution with X and the two are (in the worst case scenario) correlated. We now want to see how far real realizations Z are from $Z^\star$, taking also Y into account. Again, to evaluate the goodness of the extractor we use the variational statistical distance between real and ideal cases, namely

$$d_V((Z,Y),(Z^\star,Y^\star)) = \frac{1}{2}\min_{P_Y^\star}\left(\sum_{Z,Y}|P_{ZY}(z,y) - P_{Z^\star}(z)P_{Y^\star}(y)|\right) \tag{5.2}$$

where the ideal distributions are chosen such that Z and Y are independent variables, namely: $P_{Z^\star Y^\star}(z,y) = P_{Z^\star}(z)P_{Y^\star}(y) = \dfrac{1}{|\mathcal{A}_Z|}P_{Y^\star}(y)$. Any distribution of $Y^\star$ is fine, as long as it is independent from the one of $Z^\star$: $P_{Y^\star}$ has no meaning in reality and is

used only in the operative definition of Eq. (5.2). In the end, the job of the extractor consists not only in distilling the sequence of numbers (and thus removing the self-correlations inside a string of values of X), but also in removing correlations between X and Y so to make Z independent from Y: in this way there is no information leak to a potential eavesdropper.

With quantum side information

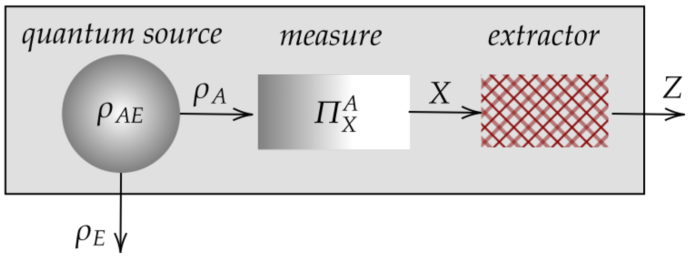


Figure 5.3: Quantum-side-information source extractor.

The idea here is that we can bound the information shared with the environment of ρ_E by measuring the state ρ_A. The state after knowing the output of the black box z will be (as expected) a classical quantum state in the form

$$\rho_{ZE} = \sum_{z \in A_Z} P_Z(z) \left|z\right\rangle \left\langle z\right| \otimes \rho_E^z \tag{5.3}$$

where ρ_E^z is the state of E *(quantum) conditioned* on the outcome of Z.
The variational statistical distance will be written in this case as

$$d_V(\rho_{ZE}, \rho_{ZE}^\star) = \frac{1}{2} \min_{\sigma_E} \mathrm{Tr} \left|\rho_{ZE} - w_Z \otimes \sigma_E\right| \tag{5.4}$$

where w_Z is a uniform superposition of i.i.d. variables (we don't need to put a star here),

$$w_Z = \frac{1}{|\mathcal{A}_Z|} \sum_z \left|z\right\rangle \left\langle z\right| . \tag{5.5}$$

Again, it is worth noticing that in the best case scenario there is no correlation between the environment and the extractor's output. If that is the case, we can write the ideal joint distribution of Z and E as a separable state and the environment state is independent from Z, $\rho_E^z \equiv \rho_E$.

2 Extractors examples

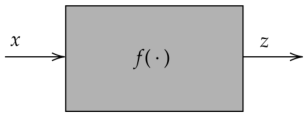


Figure 5.4: A simple deterministic function block.

Deterministic extractors

Deterministic extractors are a class of extractors that can be modelled by a function $f(X, \theta)$: given an input X and a set of parameters θ (Fig. 5.4),

$$f : \mathcal{A}_X \to \mathcal{A}_Z \cup \{\perp\}, \tag{5.6}$$

where we indicate as $\{\perp\}$ the option of discarding some input values.
Deterministic extractors differ from seeded ones since they do not require any initial true randomness seed. Examples of this type of extractors include:

— **Huffman (source) coding**. It is commonly used for lossless data compression based on the particular statistics of the source.

— **Von Neumann procedure**. As we have seen in Section 3.1, this procedures is capable of producing independent bits from a biased sequence.

— **Hashing**. It is also possible to use a cryptographic hashing functions designed on P_X as a randomness extractor[1] to generate an uniform output given a source P_X.

Since we do not care about the particular distribution P_X used to generate X, we want an extractor that works for all class of distributions, i.e. $\forall\ P_X \in \mathcal{P}_X$ where $\mathcal{P}_X = \{P_X : H_{\min}(x) \geq k\}$ (lower bounded min-entropy) and

$$H_{\min}(x, y) = \sum_y P_Y(y) \max_x P_{X|Y}(y|x) \tag{5.7}$$

It can be proven, as mentioned in [Shaltiel(2011)], that there are families of sources do not allow deterministic extraction. Historically, this led to the notion of seeded extractors that we describe in the following section.

Seeded extractors

Seeded-extractors attempt to extract pure randomness from one weak source, using an **additional number of truly random bits**, called **seed**. Obviously, this is

[1] A cryptographic hash function (CHF) is a mathematical algorithm that maps data of an arbitrary size (often called the "message") to a bit array of a fixed size (the "hash value", "hash", or "message digest").

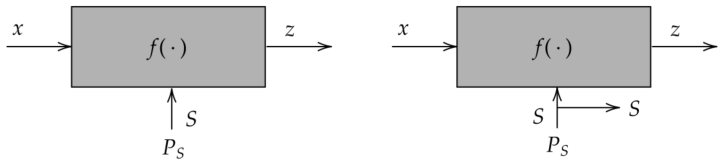


Figure 5.5: Seeded extractor on the left, strong-seeded extractor on the right.

interesting only when the length of the seed is smaller than both the length of the weak source X and the length of the output Z, i.e. **when the randomness we produce is greater than the one we consume**: in other words, from a long unbalanced sequence and a short balanced one we wish to create a long balanced sequence.
In formulas,

$$f : \mathcal{A}_S \times \mathcal{A}_X \to \mathcal{A}_Z \cup \{\perp\}. \tag{5.8}$$

A seeded-extractor is said to be **strong** if its output is *almost* (ε close, in the statistical distance sense) independent of the seed (both S and Z have uniform, independent distributions). In particular, the ideal case is the one in which both the seed and the output of the extractor follow a uniform distribution:

$$P_{Z^\star S^\star}(z, s) = \frac{1}{|\mathcal{A}_Z|} \cdot \frac{1}{|\mathcal{A}_S|}$$

Trevisan extractor

Trevisan proposed in [Trevisan(2001)] an approach to construct an m-bit extractor from any (classical) 1-bit strong extractor. Trevisan's extractor has a number of important theoretical advantages. First, it is **secure against a quantum adversary**. Second, the seed length is polylogarithmic in the length of the input[2]. Third, it can also be proven to be a strong extractor with certain modifications on the security parameters. However, a real-life implementation of this important extractor was never reported in the literature.
This extractor uses a **binary seed** $\mathcal{A}_S = \{0,1\}^{\ell_S}$, where ℓ_S is the length of the seed, and a **strong 1 bit extractor** with seed $\mathcal{A}_{S'} = \{0,1\}^{\ell_{S'}}$. The latter is a machine that works with a shorter seed than the previous one ($\ell_{S'} \ll \ell_S$) and generates only a single truly random bit $Z \in \mathcal{A}_Z = \{0,1\}$ close to the ideal case (independent on S'). We suppose that we can buy it somewhere and trust it. To understand this binary extractor we must introduce a few notions first.

[2]The case of a simple concatenation of the outputs of a 1-bit extractor (with seed of length $\ell_{S'}$) applied m times to the same input with different (independent) seeds needs a total seed of length $\ell_S = m\ell_{S'}$. Trevisan shows how to do this using only $\ell_S = \text{poly}(\ell_{S'}, \log m)$ bits of seed.

Definition 2.1 ($(\ell, \ell', m, \delta)-$design)**.** A $(\ell, \ell', m, \delta)-$design is a family of m subsets $S_1, \ldots, S_m$ of the sequence $\{1, \ldots, \ell\}$ such that $|S_i| = \ell' \ \forall i$ and $|S_i \cap S_j| \leq \delta \ \forall i \neq j$ (minimal overlapping).

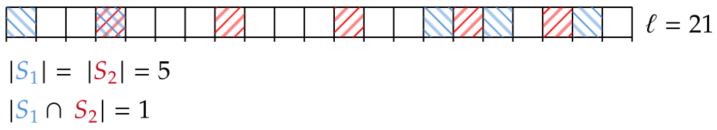


Figure 5.6: A $(21, 5, 2, 3)$-design example with $|S_1 \cap S_2| = 1 \leq 3$.

Definition 2.2 (Weak design)**.** A family of sets $S_1, \ldots, S_m$ is a weak (t, r)-design if:

— For all i, $|S_i| = t$.

— For all i, $\sum_{j=1}^{i-1} 2^{|S_i \cap S_j|} \leq rm$.

Basically for a weak design a relaxed δ boundary is set on subsystems overlap - for more information see [Raz *et al.*(2002)Raz, Reingold, and Vadhan].

Then, the Trevisan extractor consists of constructing an **m-bit extractor** from a **one-bit extractor** and a **weak design**. Let's formalize it in a proposition.

Proposition 2.1 (Trevisan extractor)**.** Let $f(s', x)$ be a strong 1-bit extractor for a source X and

$$g\left(s', x\right) \equiv \text{Concat}\left(f\left(s'_1, x\right), f\left(s'_2, x\right), \ldots, f\left(s'_m, x\right)\right),$$

where $s'_i \equiv \text{Concat}\left(\{s_j\}_{j \in S_i}\right)$ and $\{S_i\}_{i=1,\ldots,m}$ belong to a $(\ell, \ell', m, \delta)-$design. Then, g is a strong m-bit extractor for x.

Proof. Omitted □

The value of δ of the definition of the design enters the proof of this result in the computation of the variational statistical distance. Intuitively, the smaller its value, the better the performances are: in the limit of $\delta \to 0$, we are facing m disjoint sets (same as having many different input seeds of length ℓ') which means we are going towards the ideal case.
If $H_{\min}\left(x\right) \geq k$ (lower-bounded min-entropy) for a variable $x \in \{0,1\}^n$ we obtain a strong extractor asymptotically (in the number of single bits extracted $n \to \infty$) with

$m < k$ and $m \sim k^{1-\alpha}$ (where $\sim$ indicates the asymptotic equivalence, $\alpha = \alpha(\delta)$) and

$$\ell_S = \mathcal{O}\big(\log^2(n) / \log(k)\big).$$

That means that the longer the input sequence, the longer the seed must be to have a good extractor (even though the growth is just logarithmic); on the other hand, if the input min-entropy gets large (i.e. the input sequence has already a good level of secure randomness), a shorter seed is sufficient.
It is worth highlighting once again that the Trevisan extractor can be proven to be **robust to classical/quantum side information**.

As for the quantum case, with respect to Fig. 5.3, we assume a potential attacker to have access to the side information ρ_E, while the user can see the output Z and the seed S. We can give an estimate of the extractor security in term of distance from the ideal case:

$$d_V(\rho_{SZE}; \omega_S \otimes \omega_Z \otimes \rho_E) = \frac{1}{2} \min_{\sigma_E} \mathrm{Tr} \left| \rho_{SZE} - \omega_S \otimes \omega_Z \otimes \sigma_E \right|$$

Please note that $d_V = 0$ if the system provides no output (in practice E and S are completely independent).
In general, the relations between seed length and output required leads to the necessity of resolving a **trade-off** between the entropy of the source and the amount of randomness we can extract from X, $H_{\min}(X|E)$. In Table 1 we plug various weak designs and 1-bit extractors in Trevisan's construction in order to obtain concrete extractors: let $\ell_Z = \log_2 |Z|$ (where Z is the set of values z can assume), ℓ_X the length of the extractor input and $\ell_S = H(S)$. Then, for $0 < \gamma < \alpha \leq 1$, $\frac{1}{2} < \beta < 1$ the following holds:

$H_{\min}(X\|E)$	ℓ_Z	ℓ_S required	Note
$\geq k$	$\geq k - 4\log_2(1/\varepsilon)$	$\mathcal{O}\big(\log_2^3(\ell_X)\big)$	optimized output length
$\geq \ell_X^{\alpha}$	$\geq \ell_X^{\alpha-\gamma}$	$\mathcal{O}(\log_2 \ell_X)$	optimized seed length
$\geq \alpha \ell_X$	$\geq \alpha - \gamma \ell_X$	$\mathcal{O}\big(\log_2^2 \ell_X\big)$	local extractor

Table 1: Input/output trade-off for Trevisan extractor; table taken from [De *et al.*(2012)De, Portmann, Vidick, and Renner].

The value $\varepsilon = \mathrm{poly}(1/\ell_X)$ is the error and upper-bounds d_V.

3 Universal Hashing families

Definition 3.1 (Hashing function)**.** A hashing function is a general function in the form $f : \mathcal{S} \times \mathcal{X} \to \mathcal{Z}$ so that $f(s, x) = z$. It generally maps a set of data into another fixed-shape structure.

Of course, in our case the roles of s and x will be the seed and source output respectively. In particular, a hashing function can be seen as a parametric function of x once we fix a specific value for the seed s. Then, considering all possible seeds, we can define a family of functions conditioned upon the seed parameter:

Definition 3.2 (Hashing family)**.** A hashing family is a set of functions

$$\mathcal{F}_{\mathcal{S}} = \{f_s : \mathcal{X} \to \mathcal{Z}\}_{s\in\mathcal{S}}\,.$$

The analytic expression of the functions is not relevant: it is just to remark that different seeds indicate different functions that define different maps.

It is also useful to define two particular subsets of the seed domain:

$$\mathcal{S}_{X\to Z} \equiv \{s \in \mathcal{S} : f_s(x) = z\}$$

is the set of all the seeds that are mapping the source output to a well-defined value of z, while

$$\mathcal{S}_{X_1,X_2} \equiv \{s \in \mathcal{S} : f_s(x_1) = f_s(x_2)\}$$

is the set of seeds that are mapping the same source output to the same (unspecified) value of z.

Definition 3.3 ($\varepsilon-$(almost) universal family of hashing functions)**.** A family of hashing functions $\{f_s\}_{s\in\mathcal{S}}$, is said to be $\varepsilon-$(almost) universal if

$$\forall\, x_1 \neq x_2,\ |\mathcal{S}_{X_1,X_2}| \leq \varepsilon \cdot |\mathcal{S}|$$

where $\mathcal{S}$ is the set of all the possible seeds. With this definition we can label a family of hashing function to be $\varepsilon-$(almost) universal if the fraction of functions that make two generic inputs collide is limited. If the seed is random, a collision does not happen very often and we can translate all the definitions in **probabilistic terms**, i.e. we can calculate which is the probability that given two generic inputs and a fixed seed, the mapping function makes them collide. Explicitly, suppose we fix a seed s and we take a function $f_s \in \mathcal{F}_{\mathcal{S}}$; then, for two inputs x_1 and x_2 we have $z_1 = f_s(x_1)$ and $z_2 = f_s(x_2)$. We can say that $\mathcal{F}_{\mathcal{S}}$ is $\varepsilon-$(almost) universal if $P(z_1 = z_2) \leq \varepsilon$. Since this happens only if the seed is in $\mathcal{S}_{X_1,X_2}$ then we can state the epsilon universality of a family as

$$P(z_1 = z_2) = \frac{|\mathcal{S}_{X_1,X_2}|}{|\mathcal{S}|} \leq \varepsilon$$

However, there is a stronger definition of universality.

Definition 3.4 (ε-(almost)strongly universal family of hashing function)**.** A family of hashing function $\{f_s\}_{s\in\mathcal{S}}$, is said to be (almost) ε-strongly universal[3] if

$$\forall\, x, z \to |\mathcal{S}_{X\to Z}| \leq \varepsilon \cdot |\mathcal{S}| \tag{5.9}$$

so the fraction of seeds that map $x \to z$ is upper bounded and

$$\forall\, x_1 \neq x_2, z_1, z_2 \to |\mathcal{S}_{x_1\to z_1} \cap \mathcal{S}_{x_2\to z_2}| \leq \varepsilon \cdot |\mathcal{S}_{x_1\to z_1}| \leq \varepsilon^2\, |\mathcal{S}| \tag{5.10}$$

also when $x_1 \neq x_2$, the same seeds that map $x_1 \to z_1$ and $x_2 \to z_2$ must be a very small amount of the total.

Let's now concentrate on the word **almost**. We start by observing that since the $\mathcal{S}_{X\to Z}$ sets are disjoint:

$$\mathcal{S} = \bigcup_z \mathcal{S}_{X\to Z} \implies |\mathcal{S}| = \sum_z |\mathcal{S}_{X\to Z}| \underset{\varepsilon-ASU}{\leq} \sum_z \varepsilon |\mathcal{S}|$$

and if we divide by $|\mathcal{S}|$ left and right we obtain that ε is lower bounded

$$1 \leq |Z|\varepsilon \Rightarrow \varepsilon \geq \frac{1}{|Z|}.$$

When the inequality is saturated (minimum value of ε), we are in the presence of a **strongly universal** hashing function (we drop the ε-almost). Why **strongly**? The condition we posed in Eq. (5.9) and Eq. (5.10) describe now a uniform mapping and a uniform pairwise mapping respectively, which implies uniform collisions: the contrary does not hold (i.e. requiring these two condition is a stronger requirement than asking just for uniform collisions[4]). In other words, for this case we have that the probability that x_1, x_2 will hash to any pair of hash values z_1, z_2 is **as if** they were perfectly random.
For our scopes, we will limit to ε-universal families, that will be used as seeded extractors.

Examples of Universal families

Let's now have a taste of how hard is to find universal families, by giving some examples.

— We consider $\mathcal{F}_\mathcal{S}$ as the set of all the functions $\mathcal{X} \to \mathcal{Z}$, so that $|\mathcal{S}| = |\mathcal{Z}|^{|\mathcal{X}|}$ (in bits: $2^{\ell_s} = \left(2^{\ell_z}\right)^{2^{\ell_x}} \implies \ell_s = \ell_z \cdot 2^{\ell_x}$). In other words, for any value in input we can find a function $f_s \in \mathcal{F}_\mathcal{S}$ that reaches any value in output.

[3] Here is "universal"$_2$, but the binary case can be generalized to any number n.

[4] A counterexample is given by the case of an injective function (a function f that maps distinct elements to distinct elements): in this case [...]

— For $\mathcal{X} = \{0,1\}^{\ell_x}$ and $\mathcal{Z} = \{0,1\}^{\ell_z}$, a universal family is the set of all matrices $A : \mathcal{X} \to \mathcal{Z}$ on the binary field. In this case, the cardinality of the seed is equal to the number of possible configurations of the matrix (given that its element can be just $\{0,1\}$ and is size is $\ell_x \times \ell_z$, it holds $2^{\ell_s} = |\mathcal{S}| = 2^{\ell_x \ell_z} \implies \ell_s = \ell_x \ell_z$).

— All Toeplitz matrices[5] over the binary field: since the number of elements to be specified is lower, we reduce to: $|\mathcal{S}| = 2^{\ell_x + \ell_z - 1} \implies \ell_s = \ell_x + \ell_z - 1$.

In all the cases just discussed there is a **common problem**: we need to have a seed longer than the sequence of random numbers we can generate at the output of our extractor, which makes it basically useless. Actually, while defining the hashing family and its universality one should take care of some details to face this issue. In fact, we could enlarge ε to reduce ℓ_s, since ε is inversely proportional to $|\mathcal{S}|$ (Definition 3.3, Definition 3.4). As usual, this translates to a trade-off, this time between the level of universality (the value of ε) and the cardinality of the function $|\mathcal{S}|$. As a matter of fact, by shortening the seed we give up on the minimum value of ε which in return reduces the uniformity requirement and allows to choose from a wider class of families.

4 Leftover Hashing Lemma (LHL)

Hashing families are a tool that we can exploit to extract randomness. In particular, the result reported in this section with the famous lemma, allows to put an upper bound to the security measure of using them as extractors. In other words, (almost) universal hashing functions are good randomnmess extractors.
First of all, it is important to define the concept of collision:

Definition 4.1 (Collision)**.** Given two random variables X, X' following the same probability distribution $P_X(x) = P_X(x')$, we define a collision the event $X = X'$ and therefore the collision probability:

$$P_{\text{coll}}(X) = P[x = x' | x \sim P_X(x), x' \sim P_X(x')] = \sum_x P_X(x) P_X(x') = \sum P_X^2(x)$$

In particular, it is straightforward to see that if $P(x) = 1/|\mathcal{X}|$ (uniform distribution) then the collision probability is equal to the uniform distribution itself.
Notice also that while studying collisions between $z = f_s(x)$ and $z' = f_{s'}(x')$ these can occur either when $x = x' \wedge s = s'$ (by definition) or when $x \neq x'$ for some $s = s'$ (non bijective hashing function).

[5] A Toeplitz matrix is a matrix in which each descending diagonal from left to right is constant.

No side information

Lemma 4.1 (Leftover Hashing)**.** A $\delta-$universal hashing family $\mathcal{F}_{\mathcal{S}}$ with a uniform distribution of the seed ($s \sim \mathcal{U}(\mathcal{S}) \equiv 1/|\mathcal{S}|$) is a strong-seeded extractor for the input x and it holds:

$$d_V((Z,S),(Z,S)^\star) \leq \frac{1}{2}\sqrt{|\mathcal{Z}|(\delta + P_{\text{coll}}(X)) - 1} \leq \frac{1}{2}\sqrt{|\mathcal{Z}|(\delta + P_{\text{guess}}(X)) - 1}$$

Proof. Firstly, we recall that $\forall (a_1, \ldots, a_M) \in \mathbb{R}^M$ the sample mean is upper bounded by the sample power:

$$\left(\frac{1}{M}\sum_i a_i\right)^2 \leq \frac{1}{M}\sum_i a_i^2 \implies \sum_i a_i \leq \sqrt{M\sum_i a_i^2} \tag{5.11}$$

Hence, for any random variables $Y, Y^\star$ such that $Y^\star \sim \mathcal{U}(\mathcal{A}_Y)$, where $|\mathcal{A}_Y| = M$:

$$d_V(Y,Y^\star) = \frac{1}{2}\sum_Y \left|P_Y(y) - \frac{1}{M}\right| \underset{Eq.\ (5.11)}{\leq} \frac{1}{2}\sqrt{M\sum_Y \left(P_Y(y) - \frac{1}{M}\right)^2}$$

But since

$$\sum_Y \left(P_Y(y) - \frac{1}{M}\right)^2 = \underbrace{\sum_Y P_Y^2(y)}_{P_{\text{coll}}(y)} - \frac{2}{M}\underbrace{\sum_Y P_Y(y)}_{=1} + \frac{M}{M^2},$$

by substitution we get

$$d_V(Y,Y^\star) \leq \frac{1}{2}\sqrt{M\left(P_{\text{coll}}(y) - \frac{1}{M}\right)} = \frac{1}{2}\sqrt{MP_{\text{coll}}(y) - 1} \ \forall\, Y \tag{5.12}$$

and we obtain a first result by setting $Y = (z,s)$ (and therefore $M = |\mathcal{Z}|\,|\mathcal{S}|$): P_{coll} grows as the random variable differs from uniformity: the variational distance is 0 for the uniform case which gives the lowest possible value of P_{coll}. We will use this result at the end of the proof. Now, consider instead two independent trials of hashing:

$$\begin{cases} x, s, z = f_s(x) \\ x', s', z' = f_{s'}(x') \end{cases}$$

Then the collision probability is:

$$P_{\text{coll}}(z,s) = P[(z,s) = (z',s')] = P[f_s(x) = f_{s'}(x') \wedge s = s'] =$$
$$= P[x = x' \wedge s = s' \wedge z = z'] + P[x \neq x' \wedge s = s' \wedge z = z']$$

Let's work out the first term. We observe that since z depends just on x and s, then $x = x' \wedge s = s' \implies z = z'$, therefore this requirement is just redundant. Moreover, x and s are independent events and thus their joint probability is just the product of their single ones. Formally, we can write:

$$P\left[x = x' \wedge s = s' \wedge z = z'\right] = P\left[x = x' \wedge s = s'\right] = P\left(x = x'\right) \cdot P\left(s = s'\right) = P_{\text{coll}}\left(x\right) \cdot P_{\text{coll}}\left(s\right)$$

As for the second term, we can exploit the rules of conditioned probability and the fact that by hypothesis the hashing family is $\delta-$universal, therefore:

$$\begin{aligned} P\left[x \neq x' \wedge s = s' \wedge z = z'\right] &= P\left[z = z' | x \neq x' \wedge s = s'\right] \cdot P\left[x \neq x' \wedge s = s'\right] = \\ &= P\left[f_s\left(x\right) = f_s\left(x'\right) | x \neq x'\right] \cdot P\left[x \neq x'\right] \cdot P\left[s = s'\right] \leq \\ &\leq \delta \cdot \left(1 - P_{\text{coll}}\left(x\right)\right) \cdot P_{\text{coll}}\left(s\right). \end{aligned}$$

We recall that the seed follows an uniform distribution $1/\left|\mathcal{S}\right|$ (by hypothesis) and thus its collision probability is also uniform. By substituting it and combining the two previous result we finally obtain:

$$P_{\text{coll}}\left(\left(z, s\right)\right) \leq \frac{P_{\text{coll}}\left(x\right)}{\left|\mathcal{S}\right|} + \frac{\delta}{\left|\mathcal{S}\right|} - \frac{\delta P_{\text{coll}}\left(x\right)}{\left|\mathcal{S}\right|} \leq \frac{P_{\text{coll}}\left(x\right)}{\left|\mathcal{S}\right|} + \frac{\delta}{\left|\mathcal{S}\right|} \tag{5.13}$$

Finally, we can combine Eq. (5.12) and Eq. (5.13) to obtain the thesis:

$$d_V\left(\left(z, s\right), \left(z, s\right)^\star\right) \leq \frac{1}{2}\sqrt{\left|\mathcal{Z}\right| \left|\mathcal{S}\right| P_{\text{coll}}\left(\left(z, s\right)\right) - 1} \implies$$

$$d_V\left(\left(z, s\right), \left(z, s\right)^\star\right) \leq \frac{1}{2}\sqrt{\left|\mathcal{Z}\right| \left(\delta + P_{\text{coll}}\left(x\right)\right) - 1} \;\; \forall \left(z, s\right)$$

Lastly, the boundary involving the guessing probability can be found by recalling that:

$$P_{\text{coll}} = \sum_X P_X^2\left(x\right) = \sum_X P_X\left(x\right) \underbrace{P_X\left(x\right)}_{\leq P_{\max}} \leq P_{\max} \underbrace{\sum_X P_X\left(x\right)}_{=1} = P_{\max} = P_{\text{guess}}$$

□

With classical side information

Now, let us consider also the presence of classical side info. Then, we are interested in evaluating the variational statistical distance like the case before but taking into account the eavesdropper as well. Without entering the whole set of calculations, the final result of the Lemma is the following.

Lemma 4.2 (Leftover Hashing - Classical side info)**.** A $\delta-$universal hashing family $\mathcal{F}_{\mathcal{S}}$ with a uniform distribution of the seed ($s \sim \mathcal{U}(\mathcal{S}) \equiv 1/|\mathcal{S}|$) is a strong-seeded extractor for the input x with classical side info e and it holds:

$$d_V\left((z,s,e),(z,s,e)^{\star}\right) \leq \frac{1}{2}\sqrt{|\mathcal{Z}|\left(\delta + P_{\text{coll}}(x|e)\right) - 1} \leq \frac{1}{2}\sqrt{|\mathcal{Z}|\left(\delta + P_{\text{guess}}(x|e)\right) - 1}$$

In particular, if $\mathcal{F}_{\mathcal{S}}$ is strongly universal it holds:

$$d_V\left((z,s,e),(z,s,e)^{\star}\right) \leq \frac{1}{2}\sqrt{|\mathcal{Z}|\, P_{\text{guess}}(z|e)}$$

Proof. The proof follows from the one of the previous case. In fact, if we follow the definition of variational statistical distance we have:

$$d_V\left((z,s,e)(z,s,e)^{\star}\right) = \frac{1}{2}\sum_{z,s,e}\left|P_{Z,S,E}(z,s,e) - \frac{1}{|\mathcal{Z}|\,|\mathcal{S}|}P_E(e)\right| = (*),$$

for any distribution $P_E(e)$. Now, we can collect the latter by factorizing the joint probability $P_{Z,S,E}(z,s,e)$ and exploit the no-side-info LHL:

$$(*) = \sum_e P_E(e) \underbrace{\frac{1}{2}\sum_{z,s}\left|P_{Z,S|E}(z,s|e) - \frac{1}{|\mathcal{Z}|\,|\mathcal{S}|}\right|}_{=d_V\left(P_{Z,S|E},P^{\star}_{Z,S}\right)} \underset{Lemma\ 4.1}{\leq}$$

$$\leq \sum_e P_E(e)\frac{1}{2}\sqrt{|\mathcal{Z}|\left(\delta + P_{\text{coll}}(x|E=e)\right) - 1}$$

To conclude, by the means of Jensen inequality (since the square root is a concave function) we know that $\sum_i \sqrt{x_i} \leq \sqrt{\sum_i x_i}$, therefore we can bring the weighted average over $P_E(e)$ inside the square root and the thesis follows.
The particular case is trivial by substitution of $\delta = 1/|\mathcal{Z}|$. □

With quantum side information

We are now interested in estimating the security measure in presence of quantum side information.

Lemma 4.3 (Leftover Hashing - Quantum side info)**.** A $\delta-$universal hashing family $\mathcal{F}_{\mathcal{S}}$ with a uniform distribution of the seed ($s \sim \mathcal{U}(\mathcal{S}) \equiv 1/|\mathcal{S}|$) is a strong-seeded extractor for the input x with quantum side info e and it holds:

$$d_V\left((z,s,e),(z,s,e)^{\star}\right) = \frac{1}{2}\operatorname{Tr}\left|\rho_{ZSE} - \rho_{ZSE^{\star}}\right| \leq \sqrt[4]{\left(|\mathcal{Z}|\,\delta - 1\right)^2 + 9\,|\mathcal{Z}|\,P_{\text{coll}}(x|e)}$$

In particular, if $\mathcal{F}_{\mathcal{S}}$ is strongly universal it holds:

$$d_V\left((z,s,e),(z,s,e)^{\star}\right) \leq \frac{1}{2}\sqrt{|\mathcal{Z}|\, P_{\text{guess}}(z|e)}$$

The proof of this result can be found in [Tomamichel *et al.*(2011a)Tomamichel, Schaffner, Smith, and Renner].
It is worth noticing that the last equation states that **in the strongly universal case quantum and classical side information scenarios coincide**. Moreover, we see that the inequality in the strongly universal case is providing a stricter bound, yet the requirements to be satisfied to have it are often not worth it (especially the long seed needed). This is why we are usually satisfied with an almost-universality assumption: in order to have shorter seeds we allow for higher margins of δ.

More on collisions and LHL

Firstly, let's recap the results found so far in terms of the collision probability $P_C \equiv P_{\text{coll}}$:

Collision probability

No side info

$$P_C(X) = \sum_X P_X^2(x)$$

Classical side info

$$P_C(X|E) = \sum_e P_E(e) \sum_X P_{X|E}^2(x|e) = \sum_{e,X} \left(P_X E(x,e) / \sqrt{P_E(e)} \right)^2$$

Quantum* side info

$$P_C(X|E) = \Gamma_C(\rho_{XE}|\sigma_E) \equiv \text{Tr}\left(\rho_{XE} (\mathbb{1}_X \otimes \sigma_E)^{-1/2} \right)^2$$

Where in the quantum case $(*)$ we recall that ρ_{XE} is a classical-quantum state and $\text{Tr}(\sigma_E) \leq 1$. Once defined the collision probability, one can retrieve the collision entropy which is simply:

$$H_C = H_2 = \log_{1/2} P_C$$

Analogously, we can use a similar approach for the guessing probability:

Guessing probability

No side info

$$P_g(X) = \max_X P_X(x)$$

Classical side info

$$P_g(X|E) = \sum_e P_E(e) \max_X P_{X|E}(x|e) =$$
$$= \max_{g(\cdot)} P[g(E) = X] = \max_{g(\cdot)} \sum_X P_X(x) P[g(E) = x|X = x]$$

Quantum side info

$$P_g(X|E) = \sup_{\left\{M_E^X\right\}_{X \in \mathcal{A}_X}} \sum_X P_X(x) \operatorname{Tr}\left(M_E^X P_E^X\right) =$$
$$= \min_{\sigma_E} \min\left(q \in \mathbb{R} : \rho_{XE} \leq q\left(\mathbb{1}_X \otimes \sigma_E\right)\right)$$

Where $g(\cdot)$ is a decision function. Like before, the guessing entropy simply follows:

$$H_g = H_\infty = H_{\min} = \log_{1/2} P_g$$

Relationship between Collision and Guessing probabilities

In the **no side information case** we have that $P_C \leq P_g$ and it can be easily proved:

$$P_C = \sum_X P_X^2(x) = \sum_X P_X(x) \underbrace{P_X(x)}_{\leq P_{\max}} \leq P_{\max} \underbrace{\sum_X P_X(x)}_{=1} = P_{\max} = P_g$$

This is indeed a quite intuitive result: trying to guess and outcome of a random variable (x_1) can done by sampling another random variable (x_2) sharing the distribution with the one to guess and look for collisions ($x_1 = x_2$). For example, we could guess the outcome of a coin toss by tossing another one and betting on the outcome our own coin returned. So P_C will always be upper bounded by P_g.
It is worth also noticing that:

$$H_2 \geq H_{\min} \equiv H_\infty.$$

As for the **classical side information case**, the result is similar: what we need to do is to apply this inequality for all the values of the side information and then

average over the probability distribution of the side information itself,

$$\langle P_C \rangle_E \leq \langle P_g \rangle_E .$$

On the contrary, the result for the no side information case cannot be easily generalized to the quantum side information case. Before going on we define the purified distance, a metric for non-normalized quantum states.

Definition 4.2. Given two quantum states ρ and τ, we define their **purified distance** as

$$d_P(\rho, \tau) = \sqrt{1 - \bar{F}^2(\rho, \tau)}$$

where $\bar{F}(\rho, \tau) = \mathrm{Tr}\,|\sqrt{\rho}\sqrt{\tau}| - \sqrt{(1 - \mathrm{Tr}\,\rho)(1 - \mathrm{Tr}\,\tau)}$ is the **generalized fidelity**. We state that two states ρ, τ are $\eta-$distant if $d_P(\rho, \tau) < \eta$.

In the **quantum side information case** (important results were given from the Renner group in [Tomamichel *et al.*(2011b)Tomamichel, Schaffner, Smith, and Renner]) we have two main statements:

1. $\forall\, \rho_{XE}$, $(Tr\rho_{XE} \leq 1)\, \exists\, \sigma_E$, $(\mathrm{Tr}\,\sigma_E = 1) : \Gamma_c(\rho_{XE}|\sigma_E) \leq P_g(X|E)$, where one must notice that this inequality does not hold for ρ_E but only for σ_E. Let's prove it.

   Proof. By definition of $P_g(X|E)$, $\exists\, \sigma_E$ such that

   $$\rho_{XE} \leq P_g(X|E)(\mathbb{1}_X \otimes \sigma_E) = P_g(X|E)\,(\mathbb{1}_X \otimes \sigma_E)^{1/2}\,(\mathbb{1}_X \otimes \sigma_E)^{1/2}$$
   $$(\mathbb{1}_X \otimes \sigma_E)^{-1/2}\rho_{XE}(\mathbb{1}_X \otimes \sigma_E)^{-1/2} \leq P_g(X|E)\mathbb{1}_{XE}$$
   $$\rho_{XE}(\mathbb{1}_X \otimes \sigma_E)^{-1/2}\rho_{XE}(\mathbb{1}_X \otimes \sigma_E)^{-1/2} \leq P_g(X|E)\rho_{XE}$$
   $$\left(\rho_{XE}(\mathbb{1}_X \otimes \sigma_E)^{-1/2}\right)^2 \leq P_g(X|E)\rho_{XE}$$
   $$\mathrm{Tr}\left(\left(\rho_{XE}(\mathbb{1}_X \otimes \sigma_E)^{-1/2}\right)^2\right) \leq P_g(X|E)\underbrace{\mathrm{Tr}\,(\rho_{XE})}_{\leq 1}$$
   $$\Gamma_c \leq P_g(X|E)$$

   □

2. $\forall\, \rho_{XE}\, (\mathrm{Tr}\,\rho_{XE} \leq 1)\, \exists\, \rho'_{XE}$, $\eta-$close (in purified distance) to ρ_{XE} such that

   $$P_C(X|E) = \Gamma_c(\rho'_{XE}|\rho'_E) \leq P_g(X|E)\left(\frac{1}{\mathrm{Tr}\,\rho_{XE}} + \frac{2}{\eta^2}\right)$$

   If we enlarge the radius around the state ρ_{XE} where we are looking for ρ'_{XE}, i.e. allow to search on a larger set, we get a tighter bound on the collision probability.

4.1 LHL FOR QUANTUM SIDE INFORMATION

Let's consider the **universal case**, for which it holds $\delta = 1/|\mathcal{Z}|$:

$$d_V(\rho_{ZSE}, \rho^\star_{ZSE}) = \frac{1}{2}\min_{\sigma_E}\frac{1}{2}\operatorname{Tr}|\rho_{ZSE} - \omega_Z \otimes \omega_S \otimes \sigma_E| \leq \frac{1}{2}\sqrt{|\mathcal{Z}|\,\Gamma_C(\rho_{XE}|\tau_E)} \;\; \forall \tau_E.$$

In particular, we are in the condition for which we can exploit the result 1. above, hence:

$$d_V(\rho_{ZSE}, \rho^\star_{ZSE}) \leq \frac{1}{2}\sqrt{|\mathcal{Z}|\,P_g(X|E)}.$$

We can also make the bound tighter by exploiting a state ρ'_{XE} which is $\eta-$close to ρ_{XE} for which it holds as well:

$$d_V(\rho'_{ZSE}, \rho^\star_{ZSE}) \leq \frac{1}{2}\sqrt{|\mathcal{Z}|\,P'_g(X|E)}$$

Now, if we select $\rho'_{XE} = \underset{\rho' \;\; \eta\text{-close to } \rho_{XE}}{\arg\min} P'_g(X|E)$:

$$d_V(\rho_{ZSE}, \rho'_{ZSE}) \leq d_P(\rho_{ZSE}, \rho'_{ZSE}) \underset{(a)}{\leq} d_P(\rho_{XE}, \rho'_{XE}) \leq \eta$$

Where in (a) we exploited the fact that the distance between the states cannot increase during the process, and in the last two inequalities we use the purified distance. Finally, by the triangular inequality (b):

$$\begin{aligned} d_V(\rho_{ZSE}, \rho^\star_{ZSE}) &\underset{(b)}{\leq} d_V(\rho_{ZSE}, \rho'_{ZSE}) + d_V(\rho'_{ZSE}, \rho^\star_{ZSE}) \\ &\leq d_P(\rho_{XE}, \rho'_{XE}) + d_V(\rho'_{ZSE}, \rho^\star_{ZSE}) \\ &\leq \eta + \frac{1}{2}\sqrt{|\mathcal{Z}|\,P'_g(X|E)} \end{aligned} \tag{5.14}$$

Looking at Eq. (5.14) we can notice that we can put a tighter upper bound to our security measure which depends on two contributions (η and $P'_g(X|E)$) at the price of having a larger η. As usual, this is convenient only if what I pay - η - is less than what I gain - $P'_g(X|E)$. The meaning is the following: by considering a larger η we can analyze a larger neighborhood of ρ_{ZSE} and eventually find one among many distributions which can reduce the guessing probability ($P'_g(X|E)$). On the other hand, by staying close to our original state ρ_{ZSE} (small η) we are minimizing the guessing probability over a smaller set and thus we find a larger value. Therefore, there is a trade-off between the choice of η and the minimum P'_g that can be found, to be solved according to the value of $|\mathcal{Z}|$, which is setting the dominant term: a small improvement in P_g can lead to a great difference due to the value of $|\mathcal{Z}|$.
Finally, consider now the case where the hashing family is $\delta-$almost universal:

$$d_V(\rho_{ZSE}, \rho^\star_{ZSE}) \leq \frac{1}{2}\sqrt{|\mathcal{Z}|\,(\delta + \Gamma_C(\rho_{XE}|\rho_E)) - 1}$$

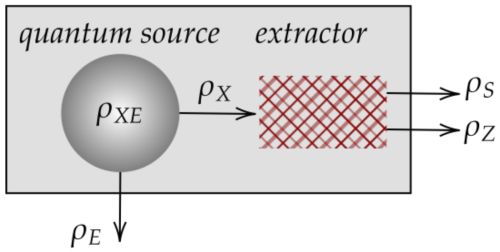


Figure 5.7: The system we are talking about in this chapter. As we can see, the state we have access to is ρ_{ZSE}.

where we need to include the real collision probability, considering ρ_E as the real side info state. Also in this case, we consider a state $\rho'_{XE} = \arg\min P'_g(X|E)$ living in the $\eta-$neighborhood of ρ_{XE}, but also another state ρ''_{XE}[6] which is $\gamma-$close to ρ'_{XE} such that:

$$\Gamma_C(\rho''_{XE}|\rho''_E) \leq P'_g(X|E)\left(\frac{1}{\operatorname{Tr}\rho'_{XE}} + \frac{2}{\gamma^2}\right)$$

and this inequality allows us to tighter the bound by working on $\Gamma_C(\rho_{XE}|\rho_E)$ by using ρ''_{XE} instead of ρ_{XE}. So, this implies:

$$d_V(\rho''_{ZSE}, \rho^\star_{ZSE}) \leq \frac{1}{2}\sqrt{|\mathcal{Z}|\left(\delta + \Gamma_C(\rho''_{XE}|\rho''_E)\right) - 1}$$

$$\leq \frac{1}{2}\sqrt{|\mathcal{Z}|\left(\delta + P'_g(X|E)\left(\frac{1}{\operatorname{Tr}\rho'_{XE}} + \frac{2}{\gamma^2}\right)\right) - 1}$$

and as we can clearly see, we can either consider the bound with the collision probability on ρ''_{XE} or the one with the guessing probability on ρ'_{XE}. In the end, by linking together all the bounds and radii we have been defining up to now, we obtain the following result thanks to the triangular inequality:

$$\begin{aligned} d_V(\rho_{ZSE}, \rho^\star_{ZSE}) \leq \gamma + \eta + \frac{1}{2}\sqrt{|\mathcal{Z}|\left(\delta + P'_g(X|E)\left(\frac{1}{\operatorname{Tr}\rho'_{XE}} + \frac{2}{\gamma^2}\right)\right) - 1} \\ + \frac{1}{2}\sqrt{|\mathcal{Z}|\left(\delta + \Gamma_C(\rho''_{XE}|\rho''_E)\right) - 1} \end{aligned} \tag{5.15}$$

In the end, since we choose ρ'_{XE} or ρ''_{XE} we obtain tighter bounds, but we will need to enlarge the distances of γ and η.

[6] A question may arise: which is the distance between ρ_{XE} and ρ''_{XE}? The answer is that they are at most $\eta + \gamma$ distant, from simple considerations on distance function properties.

4.2 Discussion on the results of the LHL

In the **no side info case**, considering a universal$_2$ hashing family, we want to generate a sequence of length $\ell_Z = \log_2 |\mathcal{Z}|$. By the means of the bounds found in the previous section, we can rewrite the variational distance

$$d_V \leq \frac{1}{2^{\frac{H_2(X)-\ell_Z}{2}+1}}$$

and let's suppose that we would like to guarantee that this distance is upper bounded by a security parameter ε. *How do we have to choose the parameters so that we can guarantee this level of security?* By simply inverting the inequality we get

$$l_Z \leq H_2(x) - 2\log_{\frac{1}{2}} \varepsilon + 2$$

so if we want more security, i.e. a smaller ε, we pay the price to output less random bits. We can make a step further: in reality we don't know the collision entropy but the min-entropy (with this we end up having a even tighter bound). Let's rewrite the inequality as:

$$l_Z \leq H_{\min}(x) - 2\log_{\frac{1}{2}} \varepsilon + 2$$

So, if ε decreases, the number of real random bits in output drops, if the $H_{\min}$ increases, the sequence of bits in output expands.

If we think to all the families of hash functions that differs from the seed s on a specific input x_1 as **codewords** in error correcting codes, asking for few collision translates to having a high **Hamming distance** between those. A $\delta-$universal family is equivalent to an error correcting code (ECC) where the length of the codewords is $n = |S|$, the number of them is $2^k = |X|$ and the code symbols are taken from $q = |Z|$. The Hamming distance should be $d_{\min} \geq n(1-\delta)$. An example can be done with Reed-Solomon codes, that gives us the possibility of having $\ell_z = \ell_s = \ell_x/2$ and this is universal with $\delta = 1/2^{\ell_z}$. Another example, $\delta = m/2^{\ell_z}$ since we are enlarging δ we get $\ell_s = (\ell_x \cdot m)/2^m$ and $\ell_z = \ell_x/2^m$.

Chapter 6

Quantum Key Distribution

Nowadays, most classical cryptography concepts rely on problems that are difficult to be worked out, or require ages to be solved. An example is the RSA encryption algorithm, that protect us while doing transactions but that can be cracked in polynomial time with a quantum computer and Shor's algorithm running on it. In general, when a quantum computer will be fully available, RSA encryption will not be secure anymore and a partial solution to this problem is **post-quantum cryptography**. The latter refers to cryptographic algorithms (usually public-key algorithms) that are thought to be secure against a crypto-analytic attack by a quantum computer. In any case, we still don't know if a classical algorithm to crack RSA exists or more in general **we are not sure that something that is secure today will be secure tomorrow**.
Nevertheless, a cryptographic method that is unconditional secure exists! This is the case of the **Vernam Cipher**, or **One-Time Pad** (OTP): an encryption technique that requires the use of a **single-use pre-shared key** that is no smaller than the message being sent. In this technique, a n-bit message x is paired with a random n-bit secret key k (also referred to as a one-time pad) to obtain the encrypted message $y = k \oplus x$. The symbol refers to modulo 2 addition, and the message can be retrieved back simply by doing $x = y \oplus k$. Shannon demonstrated that this is unconditional secure, but there is a huge weakness in this straightforward protocol: the key needs to be pre-shared between the parties and can be used only once. To carry out this delicate task one can make use of Diffie-Hellman key exchange or exploit the power and beauty of Quantum Mechanics with **Quantum Key Distribution (QKD)**, that is a secure communication method which implements a cryptographic protocol involving components of quantum mechanics. It enables two parties, usually called Alice (A) and Bob (B) to produce a shared random secret key known only to them, which can then be used to encrypt and decrypt messages.
Let us now talk about the first and most famous QKD protocol: the **BB84**, by pointing out that at the moment we are in the **discrete** and **infinite size scenario** framework, measuring the QuBit Error Rate (QBER) in the full stream of qubits.

1 BB84 protocol

The BB84 protocol is a **prepare and measure** protocol, proposed in 19**84** by **B**ennett and **B**rassard. First of all, Alice will select a string of random bits and encode each of them in one of the following four possible qubits: $|0\rangle, |1\rangle$ (Z basis) $|+\rangle, |-\rangle$ (X basis). The way she does this step is not completely casual, in the sense that she will encode zeroes half of the time into $|0\rangle$ and half into $|+\rangle$, and ones into $|1\rangle$ or $|-\rangle$ again with 50% probability each. It is important to stress that Z or X must be chosen at random (with the help of a QRNG one can generate local randomness), otherwise Eve could know the sequence of basis in advance and be **completely transparent** to the protocol (if you don't get this, stay calm and read the following pages). Bob will then receive a stream of qubits (the qubit carrier is often a photon) and since he doesn't know the base that Alice used for the encoding, he will perform random measurements, choosing either Z or X basis with equal probability.

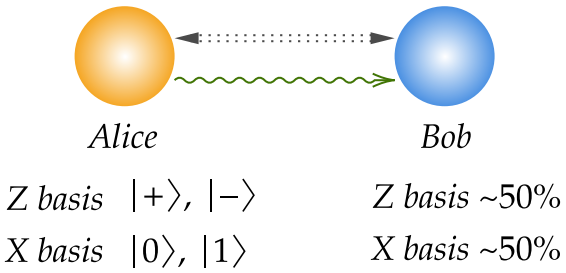


Figure 6.1: BB84 protocol

When the basis chosen by Bob is wrong, in the sense that differs from the one used by Alice, there will be no correlation between input and output. While if the two basis match, there will be in theory 100% correlation. At this point, Bob has a **raw key** k_B in his hands, that differs $k_B \neq k_A$ from Alice's original string of bits because different basis has been used (remember, the choice of basis is random). We take this into account in the next step: the **sifting phase**, where bits measured in different basis will be removed. In fact, at this point the exchange of qubits **has already happened** and the two parties will communicate the basis they used , both to encode and to measure, over an **authenticated public channel** (to avoid Man In The Middle attacks). In discrete variables this step removes also all the bits in which the photon did not arrived at the receiver due to losses in the communication, so Bob will perform the sifting only when he actually received a photon (timing is important in this protocol). It is important to notice that before this step $k_A > k_B$ and after the sifting k_A^S and k_B^S have the same length. The sequences, however, are not yet the same, and this can be due to errors in propagation (bit flips for example), or the presence of an **eavesdropper** that we will call Eve. We need to perform

information reconciliation, an error correction phase that yields $k''_A = k''_B$. At this step, the two keys are equal but not private, in the sense that there could be some information linked to the eavesdropper, and to overcome this we do **privacy amplification** $k'''_A = k'''_B$.
To understand deeply why QKD is secure, we must put some attention on the fact that we are using two non commutative basis for the encoding of the bit string. The 4 states are not orthogonal, meaning that we cannot discriminate them with any measurement, and this can be easily understood with the intercept and resend attack

1.1 Intercept and resend attack

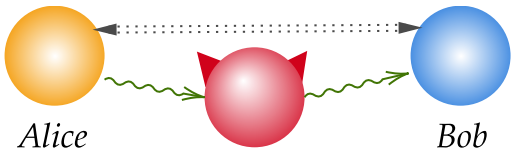


Figure 6.2: Intercept and resend attack

Eve is in between Alice and Bob, receives the photon, extracts the information and sends it to Bob. The problem for Eve is that she doesn't know the basis that Alice used to encode the information, and so she will perform the same measurement of Bob (half of the time Z basis, half X. As we will see this is not the best strategy but let's keep it for the moment). If the eavesdropper guesses correctly the basis, the measurement is faultless. If instead she measures in the wrong basis, for example basis Z when the information is in basis X, then the state will collapse to a basis Z eigenstate: $|0\rangle$ or $|1\rangle$ with 50% probability each. The measurement implies a collapse of the wavefunction into a different basis. In the sifting phase, where we retain all the bits that are results of a measurement in the same basis that Alice used to encode them, we will have 50% probability of error if the eavesdropper measures in the wrong basis. This gives the fundamental ingredient of QKD: **extracting information from a quantum system will change the system itself**, it will introduce some error, that we can measure quantitatively. In classical communication, eavesdropping is not related to errors. In the quantum world, any attempt to read a signal encoded in non orthogonal basis, translates to errors in the sifted keys. If we use orthogonal basis, the channel becomes classical. For instance, if we measure 25% of bit errors in the sifted key (1/2 of the times no error + 1/2 of the time 50% error), it means that there is no secrecy in the communication. The error is related to the security. Let's now try to evaluate the secrecy of the communication, by computing the **key rates**. Practically, depending on the error rate, the secret key rate will drop, and there is a threshold at which this key rate is zero. So, if the QBER is higher than this threshold, the transmission is aborted,

while if it is lower, the key rates will tell us that there is some secrecy left despite the presence of the eavesdropper. Finally, as it happens in any protocol, we cannot avoid denial of service attacks.

1.2 Efficient BB84 protocol

A small improvement from the last protocol springs from the observation that, since the probability to have Alice and Bob using the same basis is 50%, in the sifting phase we then lose half (a lot!) of the raw key. In the efficient version of BB84 protocol we choose the basis, both at Alice and Bob sides, with biased probabilities p_Z and $p_X = 1 - p_Z$, where $p_Z \sim 1$. Now the two basis have different roles, Z will be extracted only from Z basis measurement, while the other one is only used for checking on the eavesdropper. Eve, when performing an intercept and resend attack, will measure always in the Z basis since the key is encoded in there. However, it is destroying any information encoded in the X basis, and so this is why we use the last one for checking the presence of Eve. The advantage of doing QKD with this protocol is that we gain **almost a factor two in the key generation rate**.

1.3 Six states protocol

Another extension of the vanilla BB84, is the Six states protocol, where also the third basis in the Bloch sphere is used: p_X, p_Y and $p_Z = 1 - p_X - p_Y \sim 1$. Basically, by adding other two states in the preparation and one base in the measurement, allows to better constrain the action of the eavesdropper, and the key rate is improved by a little. Intuitively, with three basis we can completely characterize the channel, while with two we leave some freedom to a possible attacker.

2 Secret key rates

By computing secret key rates we can understand how many secure bits we have and quantify the amount of compression (driven clearly by key rates) we need to do on the raw key to obtain a private random key. For any QKD protocol, the **key rate** is defined as:

$$r = I_{AB} - I_E \tag{6.1}$$

where $I_{AB} = I(A, B)$ is the mutual information between the parties (how the two keys are correlated, and here we stop in classical communication) and $I_E = \min(I_{EA}, I_{EB})$ is the minimum between the mutual information shared between Alice (or Bob) and the environment (an eventual eavesdropper). This is related to the **direction of reconciliation** (direct or reverse): the direction in which we exchange the basis in error correction (Bob to Alice or vice-versa) actually matters a lot. There is a minimum (and not a maximum, as one may think of when trying to image the worst case scenario) because **practically** we will use direct reconciliation

(I_{AE}) or reverse reconciliation (I_{AE}) depending on the one that is minimum. Typically, reverse reconciliation is always the best and we will use $\max(I_{EA}, I_{EB})$ only when we don't know which is the minimum between direct and reverse, to consider the worst case scenario.
Quantitatively, $I(A,B) = H(A) - H(A|B)$, where

$$H(A) = -\sum_a p(a)\log_2 p(a) \quad H(A|B) = -\sum_{ab} p(a,b)\log_2 p(b|a)$$

Recalling that $p(a,b) = p(b|a)p(a)$ and

$$p(b|a) = \begin{cases} Q & b \neq a \\ 1-Q & b = a \end{cases}$$

we can deftly see that

$$H(A|B) = -\sum_{ab} p(a)p(b|a)\log_2 p(b|a) \underset{(a)}{=} h_2(Q) = -Q\log_2 Q - (1-Q)\log_2(1-Q)$$

where in (a) we used the fact that the bit is generated at random with 50% probability. From the last considerations we can write $I_{AB} = 1 - h_2(Q)$ that is the classical capacity of a classical channel. We need to communicate at least a number of bits that is the entropy of the error, otherwise we cannot correct anything. The efficiency of the error correcting code is not maximum and so in reality one puts a coefficient $f > 1$ that tells what is the efficiency of the error correcting code. In reality, I need to remove slightly more bits with respect to the ideal case.
The second term is related to leakage of information: tells what is the information I need to remove to create secrecy. There are several way to compute this error, and this depends on the attack considered:

1. **Individual attack.** The simplest attack, where Eve intercepts the qubit, performs the measurement and leaves the qubit. Here, $I_E = I(E,B)$.

2. **Collective attack.** The eavesdropper makes the qubit interact with some ancillary qubits and wait in a quamtum memory until the sifting phase. If the eavesdropper waits until the exchange of basis in the classical authenticated channel, gains more information. Here, the leak of information is related to the **Holevo bound**, namely:

   $$I_E = \chi(B,E) = S(\rho_E) - \sum_b p_b S(\rho_{E|b})$$

   where $S(\rho) = -\operatorname{Tr}(\rho\log_2\rho)$ is the Shannon entropy.

3. **Coherent attack.** The eavesdropper can do whatever she like that is in accordance with the laws of Physics: no limit to the fantasy.

2.1 Collective attack in BB84 protocol

In order to fulfill the calculations is useful to rethink the BB84 protocol as an **entanglement based** protocol

Figure 6.3: Entanglement-based protocol

Having Alice sending a state $|0\rangle$ to Bob is equivalent to measure the same basis state given the Bell state $|\phi^+\rangle$ shared among the two parties. Now, the only place where the eavesdropper can suck some information is in between the EPR source and the parties.

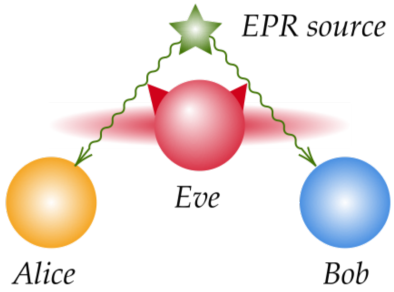


Figure 6.4: Entanglement-based protocol in the presence of the eavesdropper

Thanks to the symmetry of the protocol, the most generic attack that Eve can perform yields a density matrix state in the form of an incoherent superposition of the four Bell states:

$$\rho_{AB} = \lambda_1 \left|\phi^+\right\rangle \left\langle\phi^+\right| + \lambda_2 \left|\phi^+\right\rangle \left\langle\phi^+\right| \lambda_3 \left|\psi^+\right\rangle \left\langle\psi^+\right| + \lambda_4 \left|\psi^-\right\rangle \left\langle\psi^-\right|$$

or, with a delicate change of notation

$$\rho_{AB} = \lambda_1 \left|\phi_1\right\rangle \left\langle\phi_1\right| + \lambda_2 \left|\phi_2\right\rangle \left\langle\phi_2\right| \lambda_3 \left|\phi_3\right\rangle \left\langle\phi_3\right| + \lambda_4 \left|\phi_4\right\rangle \left\langle\phi_4\right|$$

Supposing that both Alice and Bob use the same basis to prepare and measure respectively (we look at the sifted key), the probability of having different output is the error rate, that we can measure in each basis and quantify as

$$\varepsilon_Z = \lambda_3 + \lambda_4 \quad \varepsilon_X = \lambda_2 + \lambda_4 \quad \varepsilon_Y = \lambda_3 + \lambda_2 \quad \sum_{i=1}^{4} \lambda_i = 1$$

where we put ourselves in the case of a generic Six States protocol. Intuitively, $|\phi^+\rangle$ and $|\phi^-\rangle$ have no errors in the Z basis and the bits in $|\psi^+\rangle$ and $|\psi^-\rangle$ are perfectly anti-correlated. From quantum information theory, we know that if exists a ρ_{AB} state between A and B, it is always possible to build a purification of it (that considers all the possible correlations between A,B and E) Γ_{ABE}, where $\rho_{AB} = \mathrm{Tr}_E(|\Gamma\rangle\langle\Gamma|)$. We can build this state as

$$|\Gamma\rangle = \sum_{j=1}^{4} \sqrt{\lambda_j}\, |\phi_j\rangle_{AB} |e_j\rangle_E$$

where if we trace out the eavesdropper with an orthogonal base of it, we have ρ_{AB} back. Now,

$$S(|\Gamma\rangle\langle\Gamma|) = S(\rho_{AB}) = H(\vec{\lambda})$$

If we calculate the entropy of the purified state, and we calculate the entropy of the sub part, the two quantities are the same. Now we need to calculate $S(\rho_{E|b})$, that is the entropy on the eavesdropper side given that Bob measured a bit b:

$$\frac{\mathrm{Tr}_A \langle b|\Gamma\rangle\langle\Gamma|b\rangle}{P(b)} = (1-\varepsilon_Z)|\varphi\rangle\langle\varphi| + \varepsilon_Z |\varphi^\perp\rangle\langle\varphi^\perp|$$

This means that

$$S(\rho_{E|b}) = h_2(\varepsilon_Z)$$

Applying this into the original formula of the Holevo bound

$$I_E = H(\lambda) - h_2(\varepsilon_Z) \tag{6.2}$$

For the six-states protocol, the four error rates and the λs are uniquely determined by four equations. After doing the calculations:

$$I_E = \varepsilon_Z h_2\left(\frac{\varepsilon_Z + \varepsilon_X - \varepsilon_Y}{2\varepsilon_Z}\right) + (1-\varepsilon_Z) h_2\left(\frac{2-\varepsilon_X-\varepsilon_Y-\varepsilon_Z}{2-2\varepsilon_Z}\right)$$

The key message here is that by measuring the error rates, we have a perfect knowledge on the maximum amount of the knowledge possibly owned by eavesdropper, dictated by the laws of Quantum Mechanics. We don't know if the whole amount of information is exactly in the hands of Eve, but we actually assume it as a worst case scenario. In a simplified case where $\varepsilon_Y = \varepsilon_X = \varepsilon_Z = Q$, then

$$I_E = Q + (1-Q) h_2\left(\frac{1-\frac{3Q}{2}}{1-Q}\right)$$

This is the standard benchmark for a depolarizing channel. This corresponds also to a **universal cloning machine**: an attack of the eavesdropper that achieve exactly

I_E. The attacker performs an **imperfect cloning** of the input qubit (a perfect cloning is no possible thanks to the no-cloning theorem) and the more information is cloned (fidelity of cloning) in the ancillary qubit, the higher will be the QBER. In the original BB84 instead, an equation is missing, namely $\varepsilon_Y = \lambda_3 + \lambda_2$. Here, to overcome this issue, we will perform a maximization process on Eq. (6.2), finding the values of λ that maximize the information leakage. A simple method to do calculation is to define

$$\lambda_4 = \varepsilon_Z v$$
$$\lambda_3 = (1-v)\varepsilon_Z$$
$$\lambda_2 = (1-\varepsilon_Z)u$$
$$\lambda_1 = (1-\varepsilon_Z)(1-u)$$

and with the additional constraint $(1-\varepsilon_Z)u + \varepsilon_Z v = \varepsilon_X$ we basically have a single variable left. Plugging this and maximizing, one gets, on the BB84

$$I_E = h_2(\varepsilon) \tag{6.3}$$

the information of E is given by the binary entropy on the complementary basis. Always in this protocol:

$$r = 1 - f h_2(\varepsilon_Z) - h_2(\varepsilon_X) \tag{6.4}$$

where the first is what I need to pay for the error correction, and the second is the one that I need to pay to remove information to the eavesdropper. If $\varepsilon_X = \varepsilon_Z = Q \Rightarrow \varepsilon_Y = 2Q(1-Q)$. So, the best strategy for the eavesdropper is to introduce a different error in the Y basis. This corresponds to the concept of **phase invariant cloner**. The expression of a phase invariant cloning machine is:

$$U\left|\varphi\right\rangle_A\left|0\right\rangle_E\left|0\right\rangle_X = (1-Q)\left|\varphi\right\rangle_A\left|\phi^+\right\rangle_{EX} + \tag{6.5}$$
$$+\sqrt{Q(1-Q)}\left[\sigma_Z\left|\varphi\right\rangle_A\left|\phi^-\right\rangle_{EX} + \sigma_X\left|\phi\right\rangle_A\left|\psi^+\right\rangle_{EX}\right] Q\sigma_X\sigma_Z\left|\varphi\right\rangle_A\left|\psi^-\right\rangle_{EX}. \tag{6.6}$$

This is the best cloning fidelity that we can have compatible with the laws of QM. The best cloning machine is obtained for $Q = \frac{1}{2} - \frac{1}{\sqrt{8}}$ and for this module

$$\langle\phi|\rho_E|\phi\rangle = \langle\phi|\rho_0|\phi\rangle \tag{6.7}$$

in which the lhs term is the fidelity of E and the rhs the fidelity of A.

3 QKD in multi-dimensional spaces

We treated up to now single-qubit protocols. This protocol can be generalized to **qudits**, in d-dimensional systems. First, we need to define some general variables that will be exploited in this section.

Definition 3.1 (Generalized Pauli matrix)**.**

$$\hat{U}_{jk} = \sum_{n=0}^{d-1} \exp\left(\frac{2\pi i}{d} n \cdot k\right) |n+j\rangle \langle n| \qquad j,k = 0,\dots,d-1 \tag{6.8}$$

Definition 3.2 (Generalised Bell state)**.**

$$|\Phi_{jk}\rangle = \frac{1}{\sqrt{d}} \sum_{n=0}^{d-1} \exp\left(\frac{2\pi i}{d} n \cdot k\right) |n\rangle_A \langle n+j|_B \equiv \mathbb{1}_A \otimes \hat{U}_{jk}^{(B)} |\phi_{00}\rangle \tag{6.9}$$

$$|\phi_\infty\rangle = \frac{1}{\sqrt{d}} \sum_{n=0}^{d-1} |n\rangle_A |n\rangle_B \tag{6.10}$$

We now want to define a protocol using this elements: as in the qubit state, two protocols are given:

— **Two basis protocol**, the direct generalization of the BB84 protocol, in which we we use 2 basis for QKD (as seen before), one for the actual key, and one to check for the presence of an eavesdropper. These are eigenstates of the operators $\hat{U}_{01}$ and $\hat{U}_{10}$ (i.e. the computational basis), namely

$$\{|0\rangle \dots |d-1\rangle\} \qquad \{\text{Fourier basis}\}. \tag{6.11}$$

The important thing is that all this states are considered **mutually unbiased** (MUB), such that the square of the magnitude of the inner product between any basis state equals the inverse of the dimension: $|\langle\varphi_j|\psi_k\rangle|^2 = 1/d$. This is once again the generalization of the BB84 protocol, in which the scalar product was equal to $1/2$.

— If ***d* is prime**, we can use the eigenstate of $\hat{U}_{01}$ and $\hat{U}_{1k}$ (with the index k running up to d, for a total of $d+1$ basis). The condition of d being prime assures (and is $\Leftrightarrow$ as it can be demonstrated) that we can define $d+1$ MUB: if this does not hold the number is $< d+1$ and most of the times it is unknown! A smaller number of basis is in principle possible, but $d+1$ is the optimal one and allows for maximal constraint of E.

This to say that in both protocols the first base is the key basis, all other basis are used for E detection: the larger number of E-detection basis we use, the more information we get on the channel, i.e. the higher stakes of detecting the presence of E are. This can be summarized in a trade-off between complexity (number of basis) and ease of implementation: the first point above shows us the minimal implementation, the second all the stages in between, up to the optimal case of $d+1$ basis.

For both cases we can determine the key rate: we use a standard trick where we define the error vector for the jk-basis in the form

$$\vec{q_{jk}} = (q_{jk}^{(0)}, q_{jk}^{(1)}, \ldots, q_{jk}^{(d-1)}) \qquad q_{jk}^{(t)} = \text{Prob}(a - b = t \mod d \mid j, k) \tag{6.12}$$

where a, b are the possible outputs at A and B respectively. The term $q_{jk}^{(t)}$ informs whenever we obtain a different result (no error case $t = 0$). Furthermore, the protocol can be described in an entangled fashion - since a prepare and measure process is equivalent to entanglement based protocol [1]

$$\rho_{AB} = \sum_{jk} \lambda_{jk} \left|\phi_{jk}\right\rangle \left\langle\phi_{jk}\right| . \tag{6.13}$$

We assume as usual, that Eve holds a purification of the state (i.e. knows how we generated it)

$$\left|\psi\right\rangle_{ABE} = \sum_{jk} \sqrt{\lambda_{jk}} \left|\phi_{jk}\right\rangle_{AB} \left|e_{jk}\right\rangle_{E} . \tag{6.14}$$

We have now a relation between the λ_{jk} and the $q_{jk}^{(t)}$, namely

$$\lambda_{jk} = \frac{1}{d} \left(\sum_{n} q_{1n}^{(n_j - k \mod d)} + q_{01}^{(j)} - 1 \right) ; \tag{6.15}$$

this is a $1:1$ correspondence between the two: if we know all the error rates in all the $d+1$ basis we know all the λ_{jk}: this resembles what we saw with the six-state protocol. For the key rate, as usual

$$r = I(A|B) - I_E = \log_2 d - H(\vec{q_{01}}) - I_E \tag{6.16}$$

where

1. $\log_2 d$ is the entropy of a random d-dimensional enconding,
2. $H(\vec{q_{01}})$ is the error correction part (the 01 basis is the one we use for the key exchange so the error rate in the computational basis is the one we made during error correction) and
3. H is the entropy of the $\vec{q_{01}}$ probability distribution: in detail, $H(\vec{q}) = -\sum_t q^{(t)} \log_2 q^{(t)}$.

For the last term I_E, i.e. the information on E, depends on which protocol (how many basis) we are using:

$$I_E = \chi(\rho_E) = S(\rho_{AB}) - \sum \rho(b) \rho_{E|b} = H(\vec{\lambda}) - h(\vec{q_{01}}). \tag{6.17}$$

[1] Due to the symmetries of the protocol, the correlations can be described by means of a matrix that is diagonal in the basis state

This term is completely determined in the $(d+1)$-basis case, where we can get the lambdas by means of Eq. (6.15), i.e. by measuring the quber in all the basis; this does not hold for the 2-basis case, where instead (as for the BB84 protocol) we have much more variables with respect to the quber that we measure. In this case in fact we have a lot of freedom to choose the lambdas in order to maximize the expression in Eq. (6.17). In summary, for the 2-basis state (which we remind is the straightforward generalisation to the BB84 protocol)

$$I_E = H(\vec{q}_{10}) \tag{6.18}$$

which means that the information on the eavesdropper is related to the entropy of the complementary basis while

$$r = \log_2 d - fH(\vec{q}_{01}) - H(\vec{q}_{10}) \tag{6.19}$$

where f is the efficiency of the error correction. The term $fH(\vec{q}_{01})$ is the error correction part (related to the error in the computational basis), while $H(\vec{q}_{10})$ is the privacy amplification part, related to the error in the complementary basis.
As a simple example, let's consider a depolarizing channel where all the errors are equal we have

$$\vec{q} = (1-Q, \frac{Q}{d-1}, \dots, \frac{Q}{d-1})$$

where $(1-Q)$ is the no-error rate and always in this case the entropy is

$$H(\vec{q}) = -(1-Q)\log_2(1-Q) - Q\log_2\frac{Q}{1-Q} \equiv h_d(Q)$$

which can be seen as generalization of the binary entropy formula in the d dimensional case. From this we can get the **threshold error rate**: we did not discuss it in the previous lectures but we introduce it as a useful concept. Basically the threshold error rate is the error error rate above which the key rate is > 0: this case is the only one for which we can have QKD in the first place. If we consider $f = 1$ (perfect error correction) we get the values in Table 1: the 11% threshold is the one above which we cannot have QKD! Table 1 shows some interesting patterns: fixing

d	2-basis	$(d+1)$-basis
2	11%	12.62%
3	15.9%	19.14%
4	18.9%	23.2%

Table 1: Threshold error rate percentages for $f = 1$ and $r_{\text{BB84}} = 2 - h_2(Q)$.

the dimension, from 2 to $(d+1)$ basis we increase the threshold. This is e.g. the

case for the vanilla BB84 and the six-state protocol, where we have higher tolerance to noise since in this case higher error rate are compatible with QKD: intuitively as we said, with more basis we have more knowledge of the channel and are thus able to constrain more the possibilities of E. The same happens if we increase the dimension, since we increase the number of bits for each photon (each photon carries in this case not just a single bit but $\log_2 d$ of information). By increasing the dimension however we also increase the noise, so in reality this is more of a trade-off.

4 Continuous variables QKD protocol

We now want to use infinite dimensional systems: in the case of photons this is defined by using the photon number basis (from 0 to $+\infty$). For a given coherent state $|\alpha\rangle$ the basic QKD based on continuous variables is of the form

$$A\,|\alpha\rangle \longrightarrow B : \tag{6.20}$$

on the other side, Bob's site can perform a homodyne or heterodyne measurement. Alice needs to prepare coherent states $|X + iP\rangle$ with a random distribution: these are prepared according to a gaussian distribution with 0 mean and $\tilde{V}$ variance, $\mathcal{N} \sim (0, \tilde{V})$. This is equivalent in the P vs. X plane to sample points in a circle of radius $\tilde{V}$ (which is concentric to the one representing the intrinsic variance of the vacuum) according to this same gaussian distribution.
Why is it secure? Once again, the security here lies in the **indistinguishablility** of states that are orthogonal: coherent states prepared with this protocol have this property.
The state received by Bob can be written in the form

$$\rho_B = \frac{1}{2\pi\tilde{V}} \int dx \int dp \exp\left(\frac{x^2}{2\tilde{V}}\right) \exp\left(\frac{x^2}{2\tilde{V}}\right) |X + iP\rangle \langle X + iP| \tag{6.21}$$

in which we can see the coherent state (the product $|\cdot\rangle\langle\cdot|$) prepared with some gaussian probabilities with variance $\tilde{V}$ (the product of the two exponents).
Assuming an homodyne measurement, we can calculate the variance measured by Bob:

$$V_B = \langle \hat{X}_B^2 \rangle_{\rho_D} - \underbrace{\langle \hat{X}_B \rangle_{\rho_D}^2}_{=0} \underset{(a)}{=} 4\tilde{V} + 1 \equiv V_{\text{mod}} + 1 \tag{6.22}$$

where in (a) we used $\hat{X} = \hat{a} + \hat{a}^\dagger$ (we ignored her the normalization conditions) and remembered that $\hat{a}\,|\alpha\rangle = \alpha\,|\alpha\rangle$ (the coherence states are eigenstates of the annihilation operators). The 1 in Eq. (6.22) is what is called **shot noise**, the intrinsic noise of the vacuum and the coherent state. In a classical state, this term will be 0 because all the present variance (uncertainty) would come from the preparation: here we have a convolution of the preparation and the intrinsic variance of the

coherent state.[2]In short, this relation shows how the uncertainty on Bob's measure is influenced by both the uncertainty due to the preparation and the intrinsic noise of the quantum state.
In reality however, every channel will have losses and noise: if we model those imperfections with a parameter ξ Eq. (6.22) becomes

$$V_B = \begin{cases} \eta V_{\text{mod}} + 1 + \xi \text{ (homodyne)} \\ \frac{\eta}{2} V_{\text{mod}} + 1 + \frac{\xi}{2} \text{ (heterodyne)} \end{cases}, \tag{6.23}$$

where *eta* is the transmission of the channel (the gaussian state is transformed because of the losses into a re-scaled gaussian state by a factor $\sqrt{\eta}$) and *xi* is also called **excess noise**, which is not of quantum nature but due to technical noise (background, channel, electronics). The factor 2 in the heterodyne case comes from the BS that introduces and "extra" loss.
It is important to stress that in case of discrete variables losses come from not-detected photons (that in turn correspond to bits that get thrown away): in continuous variable however we have always signal and losses cannot be removed simply by the sifting process (actually the sifted string in the heterodyne case is as long as the original key!).
Now, Bob will perform hetero- or homodyne measure: for the former, B will perform the measure by switching between basis (e.g. x and z in case of the BB84) while for the latter we have a single measurement basis (and that is why it also called **no-switching protocol**). At this point, regardless of the measurement, there will be some correlation between the X and P prepared by Alice (which she knows perfectly!) and those measured by Bob: these are called **quadrature correlations** and are fundamental for a precise key rate calculation.

The key rate is given by

$$r = (1 - \nu)(\beta I_{AB} - I_E) \tag{6.24}$$

where the first factor is related to the so called **parameter estimation** (we neglect it in the discrete case) and the parameter β is the **efficiency of the error correction** ($\beta = 1$ is the optimal case), the analogous to f of the discrete case.
How we evaluate this two parts? Starting from the mutual information,

$$I_{AB} = I(A|B) = H(B) - H(B|A) \tag{6.25}$$

which is exactly the same as for the discrete case, and for the entropy of a gaussian

[2]The advantage of choosing the normalization condition for $\hat{X}$ explained before is that for those the shot noise is equal to 1: analogously the factor in front of $\tilde{V}$ would change.

variable we have

$$H(A) \underset{(a)}{=} -\int dx p(x) \log_2 p(x) = \tag{6.26}$$

$$= \frac{1}{\sqrt{2\pi\nu}} \int dx \exp\left(-\frac{(x-x_0)^2}{2\tilde{V}}\right) \log_2 \left(\frac{1}{\sqrt{2\pi\nu}} \exp\left(\frac{(x-x_0)^2}{2\nu}\right)\right) \tag{6.27}$$

$$= \frac{1}{2} \log_2 \left(2\pi e \tilde{V}\right) \tag{6.28}$$

where in (a) we used Shannon's definition for continuous variables, replacing the sum with an integral. To compute $H(B|A)$ we know that the variable Y on the B side is related to the classical variable X through $Y = \eta X + Z$ where Z is some gaussian noise thus

$$H(B|A) = H(\eta X + Z|X) \underset{(a)}{=} H(Z|X) \underset{(b)}{=} H(Z) \tag{6.29}$$

where in (a) we used the fact that since the entropy is conditioned on X (we know X) the entropy will depend only on Z, which is uncorrelated: step (b) follows from this latter observation.
At this point, Eq. (6.26) will become

$$I_{AB} = \frac{1}{2} \log_2 \frac{2\pi e V_B}{2\pi e V_Z} = \frac{1}{2} \log_2 \frac{V_B}{V_Z} \tag{6.30}$$

i.e. the variance of the measurement on Bob's side divided by the variance of the same side. The variance of Bob is written, as we saw in Eq. (6.23), as

$$V_B = \frac{\eta}{\mu} V_{\text{mod}} + 1 + \frac{\xi}{\mu}, \tag{6.31}$$

where $\mu = 1, 2$ for homo- and heterodyne respectively, while $V_Z = 1 + \frac{\xi}{\mu}$.

Going back to Eq. (6.24), we focus now on the information of the eavesdropper I_E, the difficult part: we will not enter into details.
Also in the continous variable version, it can be demonstrated that we can generate similar correlation to the the prepare and measure process by using an entangled state between A and B and by measuring on A's side with an heterodyne, i.e.

PM $\Leftrightarrow$ EB heterodyne measurement on Alice's side.

The simple explanation for that is that if we consider the **squeezed operator**[3] defined as

$$S(r) = \exp \frac{r}{2} \left(\hat{a}\hat{b} - \hat{a}^\dagger \hat{b}^\dagger\right) = \exp\left(\tanh \frac{r}{2}\right) \tag{6.32}$$

[3] A squeezed coherent state is a quantum state that is usually described by two non-commuting observables having continuous spectra of eigenvalues: the squeeze operator is ubiquitous in quantum optics and can operate on any state. For example, when acting upon the vacuum, the squeezing operator produces the squeezed vacuum state. The squeezing operator can also act on coherent states and produce squeezed coherent states.

we can demonstrate for the 2-modes mixed state that

$$S(r)\left|0\right\rangle\left|0\right\rangle = \sqrt{i-g^2}\sum_n (-g)^n \left|n\right\rangle_A \left|n\right\rangle_B \qquad g = \tanh r \tag{6.33}$$

which is near to an entangled state, since we have n photons on A's side and n on B's side and they are coherently summed up: for $r \to \infty$ (infinitely squeezed state) we have something near a maximally entangled state.
Moreover, Eq. (6.32) can be rewritten as

$$\exp\left(\tanh\frac{r}{2}\hat{a}^\dagger\hat{b}^\dagger\right)\exp\left(-\ln\cosh\frac{r}{2}\left(\hat{a}^\dagger\hat{a}+\hat{b}^\dagger\hat{b}+1\right)\right)\exp\left(\tanh\frac{r}{2}ab\right) \tag{6.34}$$

a useful form that is used to demonstrate

$$\left\langle\alpha\right|_A S(r)\left|0\right\rangle_A\left|0\right\rangle_B \propto \left|\tanh\frac{r}{2}\cdot\alpha\right\rangle_B, \tag{6.35}$$

i.e. a coherent state on Bob's side rescaled by a factor $\tanh\frac{r}{2}$: when we measure α on A's side, we are basically preparing a coherent state on B's side. Differently from the discrete variable protocol (in the entangled version, when B measure a state, we are certain it is also the same A has prepared), there is a rescaling of the output. Apart from this rescaling, more fundamentally, Eq. (6.35) tells us that through a 2-mode squeezing and an heterodyne measurement (which is basically a projection into a coherent state) we are preparing a coherent state on the B's side an shows thus the equivalence between a prepare and measure operation and entanglement.

Similarly to the discrete variable case, given our mixed state and the purification held by the eavesdropper,

$$\rho_{AB} \longrightarrow \left|\psi\right\rangle_{ABE} = \sum_j \lambda_j \left|\psi_j\right\rangle_{AB}\left|e_j\right\rangle, \tag{6.36}$$

when we calculate the Holevo bound

$$I_E = \chi(\rho_E) = S(\rho_E) - \sum_b p(b)S(\rho_{E|b}) = \tag{6.37}$$

$$= S(\rho_{AB}) - \sum_b p(b)S(\rho_{A|b}) \tag{6.38}$$

so all the information of E is actually contained in the correlation between the two parties, Alice and Bob (a result that holds also for discrete variables). In this equation the eavesdropper has disappeared! That implies also that if we can compute the covariance (correlation) matrix, we are able to control the information on E: this is immediate in the case of gaussian states because they are states whose Wigner function has a gaussian form and the calculations of the entropy are particularly

easy: in fact, it can be demonstrated that gaussian states are completely determined by the sole covariance matrix. This implies that we can reach a full tomography of the state by knowing only the convariance matrix.
To calculate it in this case we consider

$$\hat{X} = \left(\hat{X}_1, \hat{P}_1, \hat{X}_2, \hat{X}_2\right) \tag{6.39}$$

we can define

$$V_{ij} = \frac{1}{2}\left\langle \{\Delta\hat{X}_i, \Delta\hat{X}_j\} \right\rangle_\rho \approx \frac{1}{2} \qquad\qquad \Delta\hat{X}_j = \hat{X}_j - \langle \hat{X}_j \rangle_\rho \tag{6.40}$$

where $\{\cdot,\cdot\}$ is the anti-commutator. Tipically the mean values are 0 and so for $i = j$ Eq. (6.40) becomes

$$V_{ii} = \Delta\hat{X}_i^2, \tag{6.41}$$

i.e. the standard variance, while for $i \neq j$ we have the correlation between the two.

Coming back to Eq. (6.37), we want to compute the covariance matrix of ρ_{AB} and $\rho_{E|b}$: by measuring the correlation between the X, P quadrature on A's and B's side we can directly determine them. To do that, we make use of the **symplectic eigenvalues** of V ν_k, which happen to be the "standard" eigenvalues of the matrix $|i\Omega V|$ ($\nu_k \in \mathbb{R}$ since Ω is symmetric) where

$$\Omega = \begin{pmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \end{pmatrix}. \tag{6.42}$$

Once we have that, the entropy of ρ_{AB} is given by

$$S(\rho_{AB}) = \sum_k g(\nu_k) \tag{6.43}$$

where

$$g(x) = \frac{x+1}{2}\log_2\left(\frac{x+1}{2}\right) - \frac{x-1}{2}\log_2\left(\frac{x-1}{2}\right). \tag{6.44}$$

Lastly, it can be demonstrated that any attempt of the eavesdropper will be detected in the covariance matrix: once again, at the heart of QKD lies the fact that any eavesdropping attempt will change the state itself.
The big advantage of this approach is that we can use classical detectors - for instance photodiodes (with very low electronic noise): this simplifies the implementation, since in terms of devices it is very similar to a classical computation. In principle, due to the large bandwidth of such devices, we can reach a very high key rate. A drawback however is the computational overhead of such processes: for this reason

continuous variables are preferred in short distances and high speed, contrary to what is done for the discrete variables case. The two are complementary in terms of performance, while the discrete case is better understood in terms of security.
So, **to summarize**: after the transmission (A) and the measurement (B) of the state we calculate the correlation terms for the different quadratures and retrieve the covariance matrix of the system. From the latter we can derive the symplectic eigenvalues and thus a direct expression of $S(\rho_{AB})$. From that we can compute the entropy of the state and thus retrieve an expression of the information on E.

Chapter 7

Post processing sifted keys

In this new chapter we will discuss what happens after the quantum exchange of qubits has been carried out. At this point, Alice and Bob need to create a final key to do secure communication.

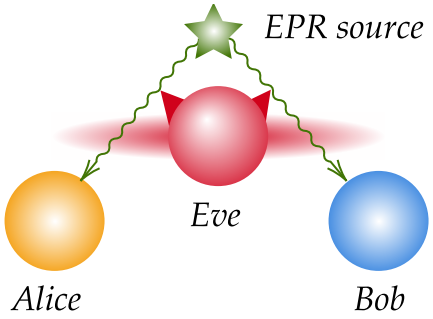


Figure 7.1: **Immagine da sostituire, caption da mantenere**. In this figure, k is an identical and uniform key, completely independent from any observation of E and from the quantum source.

The procedure that will lead to the creation of k consists of three steps:

1. Information reconciliation, leading to keys $k'_A = k'_B$
2. Verification
3. Privacy amplification, leading to $k''_A = k''_B$ independent of E

We now discuss each of them in detail in the following sections: we will call $\vec{A}$ and $\vec{B}$ the two raw keys of the two parties.

1 Information reconciliation

This part comprises of an exhange of messages m_A, m_B over the public channel (generally cheap of use), so that we **remove discrepancies** $A \neq B$ so to have two

identical yet not secret keys $k'_A = k'_B$ by leaking the **least possible** information to E. A first solution is the one proposed back in 1993 by Brassard and Salvail, the **binary protocol**.

1.1 Binary protocol

The receipe is straightforward: take the two raw keys A and B, divide them in L-bit blocks and compute the parity for each of them. The parity c of the n-th block of the two strings A and B can be computed as

$$c_n^A = \bigoplus_{i=1}^{L} A_{(n-1)L+i} \quad c_n^B = \bigoplus_{i=1}^{L} B_{(n-1)L+i}$$

Then, Alice will send her vector of parities $(c_1^A, \ldots, c_N^A)$ to Bob, which in turn will check if $c_j^A = c_j^B \; \forall j \in [1, N]$. Then, $\forall j \; : c_j^A \neq c_j^B$, we will iteratively split and perform further parity checks on the sub blocks, repeating the procedure until we locate the single bit mismatch to be corrected (or the bit is discarded).

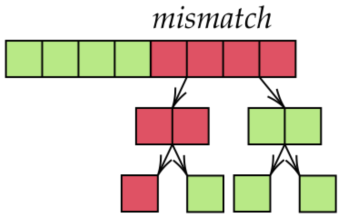


Figure 7.2: Caption

However, this simple protocol has some important drawbacks:

— The number of leaked bits is $\log_2 L \, , \forall n : c_n^A \neq c_n^B$;

— The protocol is not able to detect an even number of mismatches per block, since the parity will be the same as the $0-$errors case;

— The protocol is highly interactive (i.e, it uses the channel very much, since we are leaking info for each block and all the subdivisions of the non-matching ones.

A first way out is to choose L in a clever way given the QBER (that is known), so that is very unlikely to have multiple errors per block, i.e. $L \cdot QBER \ll 1$. Still, we cannot rule out completely the problem. Another thing that can be done is employ permutations (in a way agreed upon by both parties) of all the bits, so that mismatches belonging to the same block at a certain iteration will no longer be adjacent in the next one. At each round of permutations we correct mismatches and on average reduce the QBER (allowing for larger L). A more powerful routine for this is provided by the following protocol.

1.2 Cascade protocol

This protocol is very similar to the previous one, except from the fact that we keep **track of permutations**. In particular, if at round j we observe a mismatch in a block, this means that there were two errors among bits in that same block at round $j-1$. Following this schema, we can iteratively go back and track down all the other errors that were not identified by the binary protocol.

1.3 Linear codes protocols

A linear code $\mathcal{C}$ is a **linear subspace** in the set of binary sequences $\mathbb{B}^n$ with modulo 2 operations. DISEGNO
We can in particular consider the scenario in figure ref: the information reconciliation can be seen as the process of mapping an a $k-$bit work ($\in \mathbb{B}^k$), which in our case will be A, together with some additional information (parities) to an element of the larger space $\mathbb{B}^n$. Now, the image of the original words in $\mathbb{B}^k$ is the arrival subset of $\in \mathbb{B}^n$ that we call code, and its elements will be defined codewords. In principle, is perfect communication is held, in our case study Bob would be able to identify the original message A just by inspecting the codeword corresponding to it. However, if the channel is noisy or Eve is annoying, it could happen that the final word belonging to Bob does not belong to the code. This is however very beneficial: if Bob sees a sequence outside $\mathcal{C}$ he knows that error(s) occurred, and he can decide to request the string to be sent again or to **look for the most likely correct sequence**. In the second case, he gains the capability of error detection and correction, at the price of increasing the length of the sequence (size $k \to$ size n).
The concept of code expressed above is indeed quite intuitive: assume you are receiving a message which corrupted somehow (e.g, a piece of paper with PANEA written on it). It does mean nothing to you, so you realize an error occurred. In order to guess at your best the original content of the message, you would naturally try to put in place of the missing character all the possible letters of the alphabet, and look for the most reasonable solution. At the end of the day, you will probably decide that the most likely message is "PANDA". What did just happen? In this case, your code is nothing but the English dictionary (i.e, the set of words which are "meaningful" to you), and in particular the codewords and just the 5-chars long words in it. Since the message you received did not belong to your code, you immediately got triggered about the presence of an error and decided to associate the original message to the closest word in your code, with respect to the one you received.
We will do exactly the same in here, and dig into more detail later.
The main **parameters of the code** are:

— Hamming distance $d_H = \#$ of mismatches between two strings

— The minimum distance between a codeword and a non-codeword:

$$d_{min} = \min_{\alpha \neq \beta \in \mathcal{C}} d_H(\alpha, \beta)$$

— The maximum number of detectable errors: $d_{min} - 1$

— The maximum number of correctable errors: $\lfloor \frac{d_{min}-1}{2} \rfloor$

— The code rate: k/n (i.e, the ratio between the information transmitted and the resourced exploited

As said, we require $k < n$ and we call "redundancy digits" the values belonging to the $s = n - k$ extra dimensions.
Most of the stuff said above is rather general and applies also to non-linear codes. While coming to linear ones instead, we can represent the mapping function $g : \mathbb{B}^k \mapsto \mathbb{B}^n$ as a matrix applied to the original word:

$$\vec{c} = G\vec{b} \qquad \text{where} \qquad \vec{b} \in \mathbb{B}^k \ , \ \vec{c} \in \mathcal{C} \subset \mathbb{B}^n$$

In particular, the matrix G above is called the **generating matrix** of the code. We refer to **systematic encoding** in the case in which the generating matrix has the following form:

$$G = \begin{bmatrix} \mathbb{1}_k \\ M \end{bmatrix}$$

In this particular case, the codewords have the form $\vec{c} = \left(\vec{b}, \vec{r}\right)$, i.e. they are the concatenation of the original words together with some redundancy bits (that in our case will be the parity bits, as we will see later).
An interesting result is the following: given a non-singular matrix A, the generative matrix $G' = GA$ generates the same code as G (in formula, $\mathcal{C} = R(G)$, i.e. the rate code of G). In particular, if we use a matrix $A = P$ which just performs a permutations the rows of G, we generate an equivalent code which has the same parameters $(d_{min}, d_H, ...)$. From this it follows that **every linear code is equivalent to a systematic one**, which is given for free once we know the original info words $\vec{b}$ and the redundancy bits. Again, in our case the info words are just the A, B strings shared via the quantum channel, while the redundancies are Alice's parity bits shared with Bob on the classical public channel.
We define a **parity check matrix** any matrix $H \in \mathbb{B}^{(n-k)\times n}$ such that $\mathcal{C} = \ker(H)$, i.e.

$$\vec{c} \in \mathcal{C} \subset \mathbb{B}^n \iff H\vec{c} = 0$$

It follows that by construction $HG = \vec{0}$. Moreover, $Rank(G) = n$, $Rank(H) = n - k$. Therefore, any couple of matrices (G, H) which satisfies the previous requirement

defines a generative matrix and parity check matrix for the same code.
For the case of systematic encoding:

$$G = \begin{bmatrix} \mathbb{1}_k \\ M \end{bmatrix} , H = \begin{bmatrix} M \\ \mathbb{1}_{n-k} \end{bmatrix}$$

Since by direct application $HG = M \oplus M = \vec{0}$.

1.4 Error correction with parity check matrices

Suppose a word $\vec{c} \in \mathbb{B}^n$ is received. Then, we can compute the **syndrome** of $\vec{c}$ as $\vec{s} = H\vec{c}$. If $\vec{s} = \vec{0}$, then $\vec{c}$ belongs to the code (hence, the transmission took place non-corrupted). Otherwise, $\exists\, \vec{e} \in \mathbb{B}^n : H\vec{e} = -\vec{s}$ and therefore $c \vec{-} e \in \mathcal{C}$. We call this word the error on $\vec{c}$. Of course, there may exist multiple errors with the same syndrome (and thus multiple "possible corrections" to the received word. The **best choice is to select the error with the minimum number of errors** (i.e, the one which is closer to $\vec{0}$). At this point the best guess for the receiver will be

$$\hat{\vec{c}} = \vec{c} - \vec{e}_{min}$$

Moving back to the Alice-Bob case, assume that Alice (which has in her hands both the raw key $\vec{A}$ and its parity block values $\vec{r}_A$) encodes her codewords as $\vec{c}_A = G\vec{A}$ by the means of a systematic encoding G. This is equivalent of defining $\vec{c}_A = Concat\left(\vec{A}, \vec{r}_A\right)$. Now, Alice sends $\vec{r}_A$ to Bob on the public channel: he receives and decodes using his raw key $\vec{B}$ by calculating $\vec{c}_B Concat\left(\vec{B}, \vec{r}_A\right)$. If this word belongs to the code, he shall assume that the communication proceeded correctly and thus $\vec{B} = \vec{A}$. If not, he will find the minimum error word as described above and gets his best estimation $\hat{\vec{c}}_B = Concat\left(\hat{\vec{A}}, \vec{r}_A\right)$. Indeed, this allows him to save enormous time with respect to the binary and cascade protocols: **the whole procedure occurs in one shot**, usually with excellent results.
It is worth mentioning that since the raw keys are shared via quantum channels and the parities via classical one, we are using two different channels which could have possibly different errors and efficiencies. However, we typically employ a public channel which is both cheap and reliable, meaning that we can assume no errors on the classical communication side (so Bob receives the correct $\vec{r}_a$ without corruption).

1.5 Hashing

A similar approach can be used with hashing methods: consider a sifted key $\vec{A} \in \mathbb{B}^n$ and $\vec{s}_A = H\vec{A}$ Alice's raw key and syndromes, respectively. Now, Alice and Bob agree on a parity check matrix $H \in \mathbb{B}^{(n-k)\times n}$ (which can also be public). Then, Alice

computes her syndrome $\vec{s}_A = H\vec{A}$ and sends it to Bob, which compares it with his own syndrome $\vec{s}_B = H\vec{B}$. Observe that, by construction:

$$\vec{s}_A - \vec{s}_B = H\left(\vec{A} - \vec{B}\right) = H\left(\vec{\Delta}\right)$$

Hence, Bob can estimate $\mathbb{A}$ as $\hat{\mathbb{A}} = \mathbb{B} + \vec{\Delta}_{min}$ where $\vec{\Delta}_{min}$ is the minimum error pattern with syndrome $\vec{s}_\Delta = \vec{s}_A - \vec{s}_B$.
The approach is very similar to the previous case, but in here Alice is **transmitting the syndrome instead of the parities**. Therefore, the two parties will have to **select the most convenient method according to the size of the problem**.

HASHING	**SYSTEMATIC ENCODING**
n sifted bits	k sifted bits
$n-k$ public bits	$n-k$ public bits

Table 1: Size comparison of error correction methods

A couple of useful observations:

— The amount of bits shared on the public channel is the same, but the number of sifted bits that these methods can correct is not: while the code rate for the systematic encoder is always < 1, this is not true for the hashing. In syntheses: **hashing has better correction properties**.

— On the other hand, while the systematic encoder looks for the smallest error word in the space of raw key + parities, treating equally errors on both channels, if $\vec{s}_A$ is corrupted during transmission Bob will end up minimizing a completely wrong variable: hence, **systematic encoding has better robustness against corruptions** in the public channel.

— Note that the codes we defined in the two examples are equivalent: therefore, they can correct the same number of errors. Anyway, by taking into account also the parity bits or not in the code action we can decide the amount of sifted bits we manage to correct.

These basic concepts lead the way to more articulate procedures such as Winnow protocol and LDPC codes which will be discussed in the following sections. We now discuss the possibility to use linear codes to perform information reconciliation, as a replacement to the highly interactive protocols such as binary protocol or cascade protocol.

2 Rate vs QBER tradeoff

In order to increase the error correction capability of the code we need to, starting from the same information word length k, increase the number of redundancies or parity checks. In general, the rule is: more parity bits, more corrected errors. A concrete example is the **Singleton bound**: for any **linear code**, the minimum Hamming distance (related to the number of correctable errors) is upper bounded

$$d_{min} \leq n - k + 1$$

For example, in a systematic code we have the code word composed of all zeros and at least one that corresponds to the information $\underbrace{1000}_{k}\ldots\underbrace{0000}_{n-k}$, and we can clearly see that they can differ at most of $n - k + 1$ bits (the initial bit). Another bound, more subtle to write, is the **Hamming bound**: if the code can correct t errors, then

$$n - k \geq \log_2 \sum_{r=0}^{t} \binom{n}{r}$$

Given any code word, if the code can correct t errors then any error pattern that have at least a distance r from the code word can be corrected. The set of words that will decode to a particular code word is certainly contained in a sphere of radius $r \leq t$. If a word with errors falls outside the sphere, it can be either contained in another sphere or be completely lost in between some spheres. *How many words do I have in the sphere, i.e. words that have a distance $r \leq t$ from the code word?* All the possible ways I can choose r bits in a n bits code word. For the Singleton bound, there are no binary codes that can reach it (only non-binary codes), but a class of binary codes achieving the Hamming bound, is the one of **Hamming codes**

2.1 Hamming codes

Unfortunately, we must start by stating that this class of linear codes can correct only one error $t = 1$. On the other hand, from the perspective of the number of parity bits needed to correct errors, they ask for the least possible number of them: the decoding sphere is tightly packed in space. Every word can be either a code word or at distance 1 from a code word. Given $s = n - k$, we can construct a Hamming code with a parity check matrix H (for Hamming) whose columns are all the non-zero words of s bits. The columns will then be all the non-zero words $n = 2^s - 1$. Then, $k = n - s$, and so the parity check matrix will have n columns and and s rows.
If s parity bits are shared over a public channel, they can be observed by E, and the latter can deftly solve the linear system of equations $H\vec{a} = \vec{s}$. The solution of this is a linear space of dimensions 2^{n-s}. Every time that we leak one bit, we reduce

the set of possible sifted keys by a factor two. This is good for the legitimate entity, that has narrowed down the set of possible sifted keys to agree on a common sifted key, but this happens also on the eavesdropper side! So, if we want all the keys to be secret, we need to **reduce the sifted keys** to $n-s$ bits of security. We will need to remove s bits from the keys. The question that may arise is, *What is the best we can do in the case we don't know a prior how many errors we will correct, but we know which is the error rate (QBER) of the channel?* We would like to remove the least possible amount of bits, yet we would like to have a reasonably good probability of correcting errors. This capability depends of course on the channel error, if we know that it introduce at most t errors in any block of n bits we can say let's build a code that correct t errors. However, in general this is not the case, because the model we have for a quantum channel is the one of a **binary symmetric channel** with binary error rate $Q = QBER$: $BSC(Q)$. For each bit I share, the probability that the two bits are different is Q and is the same if it is 0 or 1, and independent on all the previous bits sent.
In this case, the Shannon bound, tells that, in the asymptotic regime $n \to \infty$ (long code words), if I want to make the probability of correcting the errors arbitrarily large

$$\lim_{n\to\infty} \frac{s}{n} \geq H(B|A) = h_2(Q)$$

where $H(B|A)$ is the binary entropy of the bit observed by B given the one observed by A, that since it is a binary channel can be rewritten as $h_2(Q)$.
In cascade protocol, we have a sort of **adaptive choice of parity bits**: we divide the sequence in blocks and compute the parity for each of them. If in the $j-$th block there are no errors, we will waste no more than one bit, otherwise we will spend $\log_2 n$ bits (iterative locating and correction procedure). We design the scheme, the choice of the block, to satisfy the requirement that the probability of having more than one error in a block is very small. If we follow a conservative approach, meaning if we **use a smaller size that we possibly could** (always referring to the error rate of the channel), we will end up with a lot of blocks with no errors that will say *no error!* and we waste only one bit of information for them: it is still a good deal. On the other hand, if we use a linear code we have **no interaction**: we expect $BSC(Q)$ and we design a code to correct this amount of error (with some margin), and **every time I will always use the same number of parity bits**. This is a problem, because if I overestimate the $QBER$ (or it is floating), all the time that I have less errors we are wasting key bits, and we will need to remove them from the key. *How do we solve this problem, that cascade does not have?*

2.2 Winnow protocol

As in the cascade protocol, we will split blocks such that it is unlikely to have more than one error in each block $QL \ll 1$, compute the parity for each of them,

and **whenever a parity mismatch is encountered** we send the syndrome of Hamming code, wasting one bit plus $2^s - 1$, so in general $L = 2^s \Rightarrow \log_2 L$ bit blocks to correct the error. As always, for the errorless blocks we waste one bit. So, from the point of view of the number of bits I share on the public channel, this protocol is totally equivalent to use Winnow or binary (cascade is a bit different because of the iterative procedure of going back on previous permutations). Winnow **does not have other interactions**, except the first check of parities. *What is the alternative to this approach of taking 'short blocks'?* The key to achieve performances close to the Shannon bound, is to use **long code words**, that on their place have the advantage of, thanks to the law of large numbers, having a smaller variance. Using short blocks I will have QL errors in each block with a large fluctuation, with long code words this does not happen.

2.3 Long code word approach

We put ourselves in the scenario of a regular channel behavior, and we recall that a long code word approach comes with a higher decoding complexity, that increases exponentially with the length of the code $\sim 2^n$. The approach pursued until 30 years ago, was to introduce strong algebraic structures in the code, such that the algorithm would find the closest code word to a given word exploiting this underlying pattern. However, nowadays this is no longer the case: finding suitable algebraic structures is a complex task and sometimes they don't even exists for certain specific codes. In the Hamming code case the length of the block must be a power of 2, and this is fixing a tight constraint on the problem. In the last decades, there has been a twist: **turbo codes**, and now **density parity check** codes. The idea is: make very long code words and find a **sub-optimal decoding scheme** which does not always give the closest code word, but **on average**.

3 Low Density Parity Check codes (LDPC)

LDPC, firstly designed by Robert Gallagher 1960, are codes whose parity check matrix H is **sparse** (with a lot of zeros as entries). They subdivide in two main classes: **regular** $(d_r,\ d_c)$ LDPC codes have exactly d_r 1s in every row and d_c 1s in every column. Recall that each row of the parity check matrix correspond to a parity bit (as many rows as parity bits) and we have as many columns as the length of the code. Of course $d_r \cdot (n-k) = d_c \cdot n$ (where $n - k = s$ the number of parity check bits and n the length of the code), so the **rate** of such code is

$$\frac{k}{n} = 1 - \frac{d_c}{d_r}$$

The second class, that is more common to use in reality, is the one of **irregular codes**, where we will randomly fill H with 1s, so that $(\bar{d}_r,\ \bar{d}_c)$ will be fixed **on**

average. For some reasons this method seems to work better, and the intuitive idea behind a more mathematical proof from Shannon, is that by initializing at random H I have more chance to fall on average in a good error correcting matrix H. Since the parity check matrix is sparse, computing the syndrome requires little computational complexity.

3.1 Iterative decoding of LDPC

Correcting errors means finding

$$\hat{A} = B \oplus \underset{\vec{e}\,:\, H\vec{e}=s_B-s_A}{\arg\min} \|\vec{e}\|_H = B \oplus \vec{e}_{min}(s_B - s_A) = \underset{a\,:\, Ha=s_A}{\arg\min} \|a - B\|_H$$

where A, B, e, $a \in \mathbb{B}^n$, s_A, $s_B \in \mathbb{B}^s$, $H \in \mathbb{B}^{s\times n}$ and the last expression means finding the closest, or the most likely a that gives the syndrome s_A given that we received B. In order to perform the decoding we build the following bipartite graph

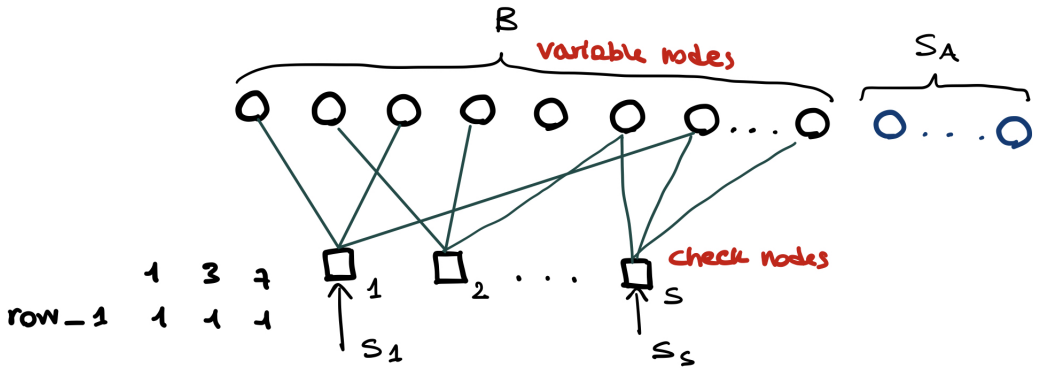


Figure 7.3: Bipartite graph.

where the **variable nodes** correspond to each received bits and compose a random vector B, and the **syndrome bits** to each parity check. We associate one **check node** to one particular row of the parity matrix, and each of them represents an equation that states that the sum of the bits (incoming lines) must be equal syndrome (or the sum of all the bits, variables and syndrome, on a check node must be equal to 0). For example, let's suppose that the matrix has row one with ones incoming from variable nodes 1,3 and 7, this means that in order to compute s_1 we need to sum bit 1,3 and 7 from the incoming word. The number of outgoing edges from each variable node is the number of ones in that column d_c. The number of incoming edges on a check node is the number of ones in that row d_r. Each of the bits b_i comes from the communication over the binary asymmetric channel (with a defined $QBER$), while the syndromes from the public channel exchange. We assume the latter to be error free. We would like to find which is the most probable

arrangement of bits, and to do this we introduce the **log-likelihood ratio**:

$$\lambda(b_i) = \ln \frac{P[b_i|0]}{P[b_i|1]} = \begin{cases} \ln 1 - Q - \ln Q & \text{if } b_i = 0 \\ \ln Q - \ln 1 - Q & \text{if } b_i = 1 \end{cases} \tag{7.1}$$

This is a measure of **how much we trust** the variable b_i we receive. If Q is below 0.5, is more likely that we receive a 0 given that a 0 was transmitted, rather than a 1 transmitted. On the other hand, for the syndromes bits, since they are error-free (with a overuse of notation, considering a binary asymmetric channel also for them we have $Q = 0$) we will have that

$$\lambda(s_i) = \ln \frac{P[s_i|0]}{P[s_i|1]} = \begin{cases} +\infty & \text{if } b_i = 0 \\ -\infty & \text{if } b_i = 1 \end{cases} \tag{7.2}$$

Update from the check node

Consider the check node c_m, then λ from c_m to b_i is

$$\lambda(c_m \to b_i) = \prod_{j,\ell} \text{sign}\lambda(b_i)\phi(\sum_{j,\ell} \phi\left(|\lambda(b_j)|\right) \quad \phi(x) = -\ln \tanh \frac{x}{2} \tag{7.3}$$

and $\phi(x) = \phi(x)^{-1}$ is a function inverse of itself. Conceptually, we are moving the Likelihood values $\lambda(b_j)$ to the ϕ domain with a non linear transformation $\phi(x)$, take the sum and then go back to the Likelihood domain. The shape of the non-linear function can be also substituted by a piece-wise approximation and all still works. The rationale behind this update is that we are computing the amount of trust (and we are doing it on every chek node) we put on the bit b_i being 0 or 1, given that at c_m we have some specific information. The sign tells us if we think b_i is either a 0 or a 1, and the second part (the one involving ϕ), tells us how convinced we are. For instance, the syndrome node has λ that is **infinity** in absolute value, and the corresponding ϕ is equal to 0, meaning that this node **is certain**, and does not give uncertainty on the knowledge of b_i. The more terms we have in the sum on c_m, the more the sum increases, and the smaller the final value gets and this corresponds to the fact that the more information we have the smaller the uncertainty will be.

Update into the variable node

This is way simpler, meaning that when we know the likelihood ratios from all the check nodes connected to a generic variable node b_i, we can simply do a sum

$$\lambda(b_i) = \sum_m \lambda(c_m \to b_i) \tag{7.4}$$

If we walk the complete graph, most of the time we end up with a solution that is overall correct. Keep in mind that is probabilistic on its core. The efficiency

of an information reconciliation protocol sending a sequence C through the public channel to help Bob recover X using side information Y can be measured using a **quality parameter** f defined as

$$f = \frac{s}{n \underbrace{(A|B)}_{h_2(Q)}} \tag{7.5}$$

where s is the number of bits we exchange on the public channel and the denominator is the minimum number of bits that we have to exchange on the public channel, written using the Shannon bound.

3.2 Code rate adaptation

We want now to alter the **code rate** in order for us to obtain a good code. We will make use of the following notation:

— $\boldsymbol{n}$, code length;

— $\boldsymbol{k}$ the code dimension;

— $\boldsymbol{s}$ the parity check bits,

for which it holds $k + s = n$. There are several ways to alter the code rate. We review them schematically in the table below ($\uparrow \cdot$ stands for increasing the quantity $\cdot$, $\downarrow$ for diminishing and $\updownarrow$ leaving unaltered).

All of them start with well designed codes: the combinations of this operations

name	s	n	k
expanding	$\uparrow$	$\uparrow$	$\updownarrow$
puncturing	$\downarrow$	$\downarrow$	$\updownarrow$
lengthening	$\updownarrow$	$\uparrow$	$\uparrow$
shortening	$\updownarrow$	$\downarrow$	$\downarrow$
augmenting	$\downarrow$	$\updownarrow$	$\uparrow$
expurgating	$\uparrow$	$\updownarrow$	$\downarrow$

allow to alter $\frac{s}{n} = 1 - R$. Two major examples are given by the puncturing and shortening actions.

— **Puncturing**. Starting from $R = \frac{k}{n}$ and applying $s' = s - \delta$ one would end up with $R' = \frac{k}{n-\delta}$: the code's rate is higher and the fraction of leaked information disclosed over the public channel is lower. We go $n' \longrightarrow n$ by adding random bits.

— **Shortening**. Since we reduce k and n, $R' < R$ and $\frac{s}{n-\delta} > \frac{s}{n}$: in the shortened code the code rate is lower and the fraction of information we leak over the public channel is larger. We go $n' \longrightarrow n$ by adding known bits.

3.3 Error verification

At the end of reconciliation, once we have corrected all the mismatches, $\hat{A} = A$ with very high probability: but what if that is not the case?

We could be simply accept it seeing the low rate of this type of security mechanism will fail (not in the sense that it will reveal sensible information to a third party but rather that the legitimate user will be not able to do anything. To further lower the chances of failure of protocol we could detect **residual errors** by computing $h = VA$, $h' = V\hat{A}$: we compute and compare the t-bit hashes over the public channel together with $V \in \mathbb{B}^{t\times n}$. If they do not match we throw the keys away: this loss is sustainable seeing the low occurrence rate.
Of curse, there is a residual probability that they still do not match given by

$$P[h = h'|\hat{A} \neq A] \leq \frac{1}{2^t} : \tag{7.6}$$

since they are universal functions, the rhs of Eq. (7.6) is the probability of a collision in the universal expansion and therefore - if i choose it randomly and uniformly - equal to 2^{-t}. If I relax my target error rate on the information reconciliation I can use verification and maybe design a code that has a larger rate and leaks less information bits.

4 Privacy amplification

Starting from the reconciled string $A = \hat{A}$, suppose E ha some information ρ_E: we wish to extract $k = f(A)$ with the following characteristics:

1. $k \sim U(\mathbb{F}^{\ell_k})$ is uniform;
2. k is independent of ρ_E.

We can write this in terms of the variational distance d_V (for and ε-unconditional security)

$$\min_{\sigma_E} \mathrm{Tr}\left[\rho_{kE} - \omega_k \otimes \sigma_E\right] \leq \varepsilon. \tag{7.7}$$

We focus on a special case in which E holds only classical information,

$$\mathbb{F}^{\ell_e} \ni E = MA \qquad\qquad M \in \mathbb{F}^{\ell_e \times n}$$

where M is known. For k on the other hand,

$$\mathbb{F}^{\ell_k} \ni k = PA \qquad\qquad P \in \mathbb{F}^{\ell_k \times n}.$$

In order for k to be uniform, starting from A that is uniform itself, the only condition that we need to impose is that P allows to reach every point in $\mathbb{F}^{\ell_e \times n}$ as in Fig. 7.4.

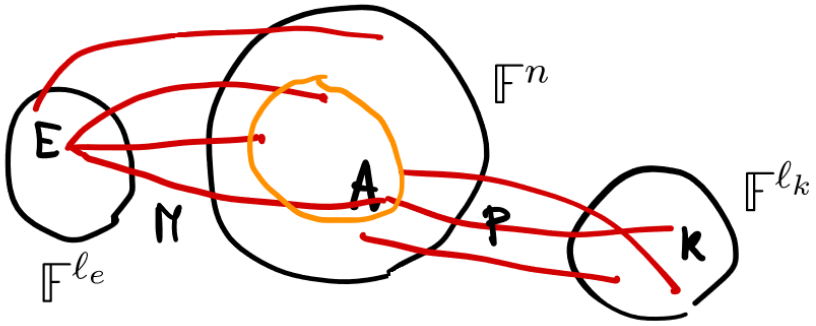


Figure 7.4: Schema for the action of privacy amplification: the maps go from A outwards.

Once the adversary learns E, what he knows is that the string A will be one point inside the golden circle in Fig. 7.4: this means that she has reduced the number of possible As from $\mathbb{F}^n$ to the size of A.
What we need is the map via P into k, so that all keys there are still reachable not only by the set $\mathbb{F}^n$ but even by its subset A.
If we denote with $\mathcal{N}(\cdot)$ the **null-space** of a generic matrix,

$$\dim(\mathcal{N}(M)) - \dim(\mathcal{N}(M) \cap \mathcal{N}(P)) = \ell^k \equiv \text{rank}(P) \implies P_k,\, P_{k|E} \sim U(\mathbb{F}^{\ell^k}).$$

What about whatever other information E has? For the **general case**, starting from A (with ρ_E unknown) we want to retrieve k through Eq. (7.7): this action reminds us of the **randomness extractors** since we have now removed the process of reconciliation. This suggests us also the solution: what is typically done here is to choose P randomly from a universal$_2$ class. We can now use the LHL that adapted to this case reads

$$\varepsilon = \frac{1}{2}\sqrt{|\mathbb{F}^{\ell_k}| \cdot P_{\text{guess}(A|E)}}.$$

The only point is now to choose properly ℓ_k since we already know how to build a universal$_2$ hashing function and we have an estimate of the min-entropy through

$$P_{\text{guess}}(A|E) = \frac{1}{2^{H_{\min}(A|E)}}.$$

5 Authentication and integrity protection

Everything that we have said up to now relies on communication over the public channel (the process of sifting, information reconciliation, error verification and privacy amplification). The public channel is by definition both readable and accessible by anyone: the attacker could in principle block all communications and chooses his

own sifted keys and his own syndrome (man in the middle attack), a situation we wish to avoid. In order for the protocol to be trusted, the public channel resources must be **authenticated**: for instance we require **unconditionally secure authentication** and **integrity protection**. Let's see how.
Given a message c, we compute its **signature**

$$s = T_k(c),$$

where $\{T_k(\cdot)\}_k$ is a ε-almost strongly universal family (we have consequently ε-unconditional secure authentication). The problem is that k must be chosen randomly and uniformly - and can be used one-time only! Also, k must be "sufficiently" long, which complicates its sharing over the public channel: if k is longer than the key we actually need do communicate, the key distribution protocol defies its purpose. If the protocol is repeated $c_1, \ldots, c_N$ with signatures $T_1, \ldots, T_N$, then it is sufficient to choose

$$t_i = T_k(c_i) \oplus k'_i$$

where $\oplus$ identifies the one-time pad operation on the signature T_k and k'_i changes every time: now the number of key bits is no longer k (which is long) but k'_i, a short OTP to the signature.

Chapter 8

Security proof for QKD

1 An entropic inequality

In this chapter we wrap up what we learned in the previous sections in order to show the security of the whole protocol and postprocessing we described and see how the choice of parameters affects the security rate.
There are multiple proofs of such argument: the one reported here is proposed in paper
DISEGNO
Considering the previous image, we are interesting in the following uncertainty metric:

$$H(X|B) + H(Z|E) \geq q \equiv \log_{1/2} \max_{x,z} \left\| \sqrt{\hat{M}_x} \sqrt{\hat{M}_z} \right\|_\infty^2$$

Where $\left\{ \hat{M}_z \right\}_z$ are the measurement operators on the Z basis (same for X). Note that if $\hat{M}_z = |z\rangle\langle z|$ and $\hat{M}_x = |x\rangle\langle x|$ (projective measurement), then it holds $\hat{M}^2 = \hat{M}$ and we get $q = \log_{1/2} |\langle x|z\rangle|^2$
It is worth also noticing that if $X = Z$ then $|\langle x|z\rangle| = 1 \implies q = 0$ (useless bound). Instead, if the two bases are mutually unbiased (eg, spin along orthogonal directions) then $|\langle x|z\rangle| = 1/d \implies q = \log_2(d)$ (as in the entropic uncertainty principle).
We can extent this concept to min/max entropies: $H_{max}(X|B) + H_{min}(Z|E) \geq q$. The role of q is the one of measuring the overlap between X and Z. The proof is made in three steps.

Step one: proof for pure density matrix

If ρ_{ABE} is pure, then by definition $H_{max}(X|B) = -H_{min}(X|AE)$. It can be proven (we will not do it) that the latter is $\geq q - H_{min}(Z|E)$.

Step two: extend to smooth entropies

Consider a state ρ' in the $\epsilon-$neighborhood of $\rho = \rho_{ABE}$ (in purified distance) that approximates the latter. It holds:

$$H^{\epsilon}_{max}\left(X|B\right)_{\rho} = \inf_{\rho'} H_{max}\left(X|B\right)_{\rho'}$$

Similarly, consider another state ρ" which takes the role of ρ' above:

$$H^{\epsilon}_{min}\left(Z|E\right)_{\rho} = \sup_{\rho"} H_{min}\left(Z|E\right)_{\rho"} \geq H_{min}\left(Z|E\right)_{\rho'}$$

Where the last inequality is a clear consequence of the definition of sup. Note: if ρ is pure, also the supremum and inferior are reached at pure states.
Considering now ρ pure:

$$H^{\epsilon}_{min}\left(Z|E\right)_{\rho} + H^{\epsilon}_{max}\left(X|B\right)_{\rho} \geq H_{min}\left(Z|E\right)_{\rho'} + H_{max}\left(X|B\right)_{\rho'} \geq q$$

Step three: extend to mixed states

Consider now a mixed state ρ_{ABE} and its purification ρ_{ABED}. Then, from the previous point:

$$q \leq H^{\epsilon}_{min}\left(Z|E\right) + H^{\epsilon}_{max}\left(X|BD\right) \geq H^{\epsilon}_{min}\left(Z|E\right) + H^{\epsilon}_{max}\left(X|B\right)$$

Where the max entropy above can not increase by restricting to the only B subsystem.
Now, applying $\hat{M}_x$ to ρ_B, consider an outcome X'. Then $H^{\epsilon}_{max}\left(X|X'\right) \geq H^{\epsilon}_{max}\left(X|B\right)$.
DISEGNO
This is indeed quite intuitive: since X' is linked to X via ρ_B, it is impossible for it to share with X more info than what ρ_B does, hence the max entropy cannot decrease.
In the end, wrapping up all the results above, one could recast:

$$H_{max}\left(X|B\right) + H_{min}\left(Z|E\right) \geq q$$

2 Privacy amplification

The following idea comes from paper. Assume to choose $P\left(\cdot\right)$ uniformly in an universal class where $P\left(U\right) = k \in \mathbb{B}^{\ell}$. Then, recollecting that Alice and Bob will use more the Z basis (key encoding) than the X basis (Eve detection), Alice will create a couple $Z \mapsto \left(U, V\right)$ where U is kept with Alice and V is sent (publicly) to Bob. In this scenario, the maximum length ℓ which allows for $\epsilon-$secrecy (i.e, $\min_{\sigma_E} \mathrm{Tr}\left|\rho_{KE} - \omega_K \otimes \sigma_E\right| < \epsilon$) is

$$\ell^{\epsilon}_{sec} \geq \sup_{Z \mapsto (U,V)} H^{\epsilon_1}_{min}\left(U|EV\right) - H^{\epsilon_2}_{max}\left(U|BV\right) - 4\log_{1/2}\epsilon_3 - 3 \text{ s.t. } \epsilon = \epsilon_1 + \epsilon_2 + 2\epsilon_3$$

The bound before is written in terms of smooth entropies, conditioned to the public information (V is public, E is what Eve knows). Observe that using the result of the second and third step of the previous section, we can use $q - H_{max}^{\epsilon_1}(X|X')$ instead of $H_{min}^{\epsilon_1}$. Moreover it holds:

$$\ell_{sec}^{\epsilon} \leq \sup_{Z \mapsto (U,V)} H_{min}^{\sqrt{2\epsilon}}(U|EV) - H_{max}^{\sqrt{2\epsilon}}(U|BV)$$

As a result, we have an upper and lower estimate of the length as a function of the neighborhood size where we approximate the density matrix.

3 Finite key analysis of efficient BB84

The following idea comes from paper. Let us recap the steps of the whole procedure:

1. Select majority and minority bases: Z for the key, X for Eve detection, probabilities $P_z \approx 1$ and $P_X = 1 - P_Z$.

2. After sifting, Alice and Bob have n_Z bits measured in the Z basis (n_X in X): on average, over n_{tot} bits sent, $n_Z = n_{tot}P_Z^2$ and $n_X = n_{tot}P_X^2$ (coincidences only if the same basis is used).

3. We recall that for this choice of probabilities the optimum strategy for Eve is to measure always in the Z basis. As a result, the expected QBER on this basis is zero, while we can estimate the one on X as:

$$\hat{Q} = \frac{1}{n_X}\sum_{i=1}^{n_X} |X_i - X_i'| \equiv \frac{1}{n_X}\sum_{i=1}^{n_X} X_i \oplus X_i'$$

   Then, if $\hat{Q} > Q_{tol}$ for a fixed tolerance parameter Q_{tol}, abort the protocol (Eve has too much info).
   Note: we indicate with $X_1, \ldots, X_{n_X}$ the values of the bits measured by Alice, with $X_1', \ldots, X_{n_X}'$ the ones of Bob (and similarly for the Z basis measures).

4. Perform (either backward of forward) information reconciliation sending s bits on the public channel (e.g., if forward, B reconstructs $\hat{Z}$ from Z, s.

5. Perform verification sharing r bits of hash ($r = \lceil \log_{1/2} \epsilon_{corr} \rceil$)

6. Perform privacy amplification with a $\mathbb{B}^{n_Z} \to \mathbb{B}^{\ell}$ universal hash.

Hence, the parameters of the protocol are:

— n_Z, n_X: the amount of bits collected in sharing

— Q_{tol}: the maximum tolerable QBER

— s: the number of bits for information reconciliation

— ϵ_{corr} (or equivalently r): the number of bits for verification

— ℓ: final length of the secure key

What do we aim to guarantee with such protocol?

1. **Correctness**: $P\left(k_A \neq k_B\right) \leq \epsilon_{corr}$
2. **Secrecy**: $\min_{\sigma_E} \operatorname{Tr}\left|\rho_{KE} - \omega_K \otimes \sigma_E\right| \leq \epsilon_{sec}$
3. **Robustness**: $P\left(abort, no-attack\right) \leq \epsilon_{rob}$

Let us have a closer look at the first ones:

1. If we exploit universal$_2$ hash to perform error verification and $r \geq \log_{1/2} \epsilon_{corr}$, then we are $\epsilon_{corr}-$correct. In fact:

$$P\left(k_A \neq k_B\right) = P\left(\hat{Z} \neq Z \ \ \& \ \ r_A = r_B\right) =$$

$$= \sum_{z,\hat{z}} P_{Z,\hat{Z}}\left(z,\hat{z}\right)\left[P\left(r_A = r_B | \hat{Z} = \hat{z}, Z = z\right)\right] \leq \sum_{z,\hat{z}} P_{Z,\hat{Z}}\left(z,\hat{z}\right)\left[\frac{1}{2^r}\right] = \frac{1}{2^r} \leq \epsilon_{corr}$$

2. Let Q be the QBER we would have measured flipping the bases roles (i.e, using X for the key). Theoretically, we would expect $Q \approx \hat{Q} \leq Q_{tol}$, where $\hat{Q}$ is the QBER on X estimated before. We can rewrite this as $Q \leq Q_{tol} + \mu$, for some value μ. Consider now a value of ℓ such that:

$$\ell \leq n\left(q - h_2\left(Q_{tol} + \mu\right)\right) - s - r - 2\log_{1/2} \epsilon_{sec} - 1$$

Where $q = \log_{1/2} \max_{X,Z\in\{0,1\}} \left|\langle X|Z\rangle\right|^2$. If the previous is satisfied, then we have $\epsilon_{sec}-$secrecy. Observe that in particular μ is chosen in a Gaussian fashion as

$$\sqrt{\ln\left(\frac{2}{\epsilon_{sec}}\right)\left(\frac{1}{n_Z} + \frac{1}{n_X}\right)\left(1 + \frac{1}{n_X}\right)}$$

Note that $\mu \to 0 \iff n_Z, n_X \to \infty$. This is indeed expected: if we use an infinite amount of bits to characterize the two bases, we can improve our knowledge secrecy. Moreover, in the boundary of ℓ above we recap that $s \sim nh\left(Q\right)$ asymptotically and in the case of a completely depolarizing channel $Q \approx \hat{Q} \approx Q_{tol}$, so for $\mu \to 0$ the previous result reduces to the one of the $\infty-$long case.

Chapter 9

Real implementation of QKD

Having perfect and stable single photons sources in reality is very hard. However, we can demonstrate that secure quantum protocols of QKD can be implemented even with attenuated coherent light (lasers): the so called **decoy states**. This idea of is simple and deep. Alice changes the nature of the quantum signal at random during the protocol; at the end of the exchange, she will **reveal which state she sent in each run**. This way, Eve cannot adapt her attack to Alice's state, but in the post-processing Alice and Bob can estimate their parameters conditioned to that knowledge. As we know, lasers generate **coherent states** (superpositions of number states)

$$|\alpha\rangle = \left|\sqrt{\mu}e^{i\theta}\right\rangle = \sum_{n=0}^{\infty} e^{-\frac{\mu}{2}} \frac{(\alpha)^n}{\sqrt{n!}} |n\rangle \tag{9.1}$$

where the parameter $\mu = \langle\alpha|\alpha\rangle = |\alpha^2|$ stands for the **intensity** (average photon number) of the laser, and this is exactly the **knob we are going to tweak** in order to change the nature of the quantum state. In other words, we will generate a laser pulse with a fixed μ for each run. The phase factor $e^{i\theta}$ is accessible if a reference for the phase is available; if not, the **emitted state is described by the mixture**

$$\rho = \int_0^{2\pi} \frac{d\phi}{2\pi} \left|e^{i\phi}\sqrt{\mu}\right\rangle \left\langle e^{i\phi}\sqrt{\mu}\right| = \sum_{n=0}^{\infty} P_\mu(n) |n\rangle \langle n| \tag{9.2}$$

with

$$P_\mu(n) = \frac{e^{-\mu}\mu^n}{n!} \tag{9.3}$$

being a *Poisson*. This comes from the fact that two equivalent component of the same density matrix cannot be distinguished, and in the absence of a phase reference the laser at intensity μ produces a *Poissonian mixture* of number states. Without making further clarifications one can audibly notice the **intrinsic weakness** of the usage of a simple[1] laser light: Eve can keep one of the photons of a multi-photon

[1] where we can have multi-photons states without any control

state, store it in a quantum memory (**we are not saying that this is possible** nowadays, but that is physically possible) and **wait** until the measurement of Bob and the respective collapse. More deeply, **photon-number-splitting** (PNS) attacks are possible due to the fact that ρ coming out from Alice's laser **commutes** with the number operator.

The existence of a reference for the phase is essential in both continuous-variable and distributed-phase-reference protocols: after all, these protocols have been designed having specifically in mind the laser as a source. On the contrary, when attenuated lasers are used to approximate qubits in discrete protocols, the phase **does not play any role**

1 Efficient BB84

We are now interested in the specific efficient BB84 protocol. We are interested in R_x, R_z sifted key rates, that they can definitely be measured during a secure communication. What **cannot be measured** instead are $R_{x,n}$ and $R_{z,n}$, the sifted key rates in the case n photons are transmitted in the quantum channel. We have the following relation

$$R_b = \sum_n R_{b,n} \quad b = x, z \tag{9.4}$$

and we can define the **error rate**

$$E_b R_b = \sum_n R_{b,n} e_{b,n} \tag{9.5}$$

where $e_{b,i}$ is the **quantum bit error rate** and the infinite size key rate

$$r = q\left[R_z(1 - f_{EC} h_2(E_z)) - R_E\right] \tag{9.6}$$

where the **rate of eavesdropper information** is

$$R_E = \sum_n I_{E,n} R_{z,n} \tag{9.7}$$

In the case of a generic prepare and measure protocol we will have that $I_{E,0} = 0$ and $I_{E,n\geq 2} = 1$, i.e. the eavesdropper has full information on the system for a multi-photon system. All multi-photon events are totally insecure. For one photon instead, $I_{E,1} = h_2(e_{x,1})$. Having found these two results and recalling that

$$\sum_{n=2}^{\infty} R_{b,n} = R_b - R_{b,1} - R_{b,0} \quad b = x, z$$

we can rewrite Eq. (9.7) as

$$R_E = \sum_n I_{E,n} R_{z,n} = I_{E,0} R_{z,0} + R_{z,1} h_2(e_{x,1}) + \sum_{n=2}^{\infty} R_{z,n} \tag{9.8}$$

$$= R_{z,1} h_2(e_{x,1}) + R_z - R_{z,1} - R_{z,0} \tag{9.9}$$

Also the expression above for the general and infinite size key rate can be recasted into

$$r = q \left[R_{z,0} + R_{z,1}(1 - h_2(e_{x,1})) - \underbrace{R_z f_{EC} h(E_z)}_{\text{pay for error correction}} \right] \tag{9.10}$$

As aforementioned, $R_{z,0}$ and $R_{z,1}$ are not known but we know for sure that is their generic form is

$$R_{b,n} = P_\mu(n) Y_{b,n} \tag{9.11}$$

where the second factor can be at most 1, it is called **yield** and stands for the **probability** that E forwards some signal to Bob for n-photon pulses. One must notice that $Y_{b,0}$ is the background rate which includes the detector dark count and other unpredictable contributions. We can make the assumption that all the error come from the single photon case

$$E_x R_x = \sum_{n=0}^{\infty} R_{x,n} e_{x,n} \geq R_{x,1} e_{x,1} \rightarrow e_{x,1} = \frac{R_x}{R_{x,1}}$$

we assume $R_{z,0}$: whenever E sees 0 photons knows everything. We are missing a piece. E can measure number of photons without introducing error (the polarization measurements commute with the number operator), so the best strategy for E would be to block vacuum events and transmit multi-photons: $Y_{b,0} = 0$, $Y_{b,n\geq 2} = 1$. Under this conditions:

$$R_z = R_{z,1} + \sum_{n=2}^{\infty} P_\mu(n) \tag{9.12}$$

So that

$$R_{z,1} \geq R_z - (1 - P_\mu(0) - P_\mu(1))$$

Expanding in Taylor series $P_\mu(n)$ for small μ and for $n = 0$ and $n = 1$ we can rewrite the quantity in the parenthesis in the last expression as $P_\mu(0) \approx 1 - \mu + \frac{\mu^2}{2}$ and $P_\mu(1) = \mu e^{-\mu} \approx \mu - \mu^2$: $1 - P_\mu(0) - P_\mu(1) = \frac{\mu^2}{2}$.
Considering a **more general case** of a a channel with losses η, then $R_z \approx \mu\eta$ and

$$R_{b,1} \geq \mu\eta - \frac{\mu^2}{2} = \mu\left(\eta - \frac{\mu}{2}\right)$$

If $E_x = E_z = Q = 0$, then

$$r = qR_{z,1} = q_\mu \left(\eta - \frac{\mu}{2}\right) \tag{9.13}$$

we can have an estimate for μ

$$\frac{\partial r}{\partial \eta} = 0 \iff \mu_{opt} = \eta \to r = \frac{1}{2} q\eta^2$$

In the end, we will need to change μ depending on the losses of the quantum channel, but the main statement is: **with lasers we can perform QKD!**

1.1 Decoy state approach

The idea of decoy states is simple and deep. Alice **changes the nature of the quantum signal** at random during the protocol; at the end of the exchange of quantum signals, she will reveal which state she sent in each run. This way, Eve **cannot adapt her attack** to Alice's state, but in the post-processing Alice and Bob can estimate their parameters conditioned to that knowledge. Suppose then we can change μ in the Poissonian Eq. (9.3): $\mu_1 > \mu_2 > \cdots > \mu_N$. Whenever Eve finds 3 photons, she does not know from which μ they come from.
PNS is different for each μ_i. Suppose $Y_0 = 0$, $Y_p =$, $Y_{n\geq 1} = 1$, then

$$R_z^\mu = \sum_n P_\mu(n) Y_n = pP\mu(1) + (1 - P_\mu(0) - P\mu(1)) \tag{9.14}$$

so if we change μ, all the values will change.

$$R_z^{\mu_j} = \sum_n P_{\mu_j}(n) Y_n \tag{9.15}$$

The more constraint I add, the best the estimate Y_n will be. In the infinite of N decoy states: $R_z = \eta\mu$, $R_{z,1} = \mu e^{-\mu}\eta$ and

$$r = \eta\mu \left(e^{-\mu}(1 - h_2(Q)) - f_{EC} h_2(Q)\right) \propto \eta \tag{9.16}$$

and now the optimal μ, calculated again by extremizing the key rate, is not anymore dependent on η:

$$\mu_{opt} = \frac{1}{2}\left(1 - \frac{h_2(Q)}{1 - h_2(Q)}\right) \tag{9.17}$$

and this is **similar to the single photon case**. Convenient! The only complexity is in the random modulation of the intensity over polarization.
We can now demonstrate that three intensity levels, i.e. **three mean photon** coefficients $\mu_1 > \mu_2 > \mu_3$ are enough to achieve a good performance. Modulating the intensity we have info on parameters.

$$r^{\mu_j} = q\left[R_{z,0}^{\mu_j} + R_{z,1}^{\mu_j}(1 - h_2(e_{x,1})) - R_z^{\mu_j} f_{EC} h_2(E_z)\right] \tag{9.18}$$

and our final estimate will be

$$r = \sum_j P_{\mu_j} r^{\mu_j} \tag{9.19}$$

We recall that

$$R_z^{\mu_j} = \sum_n P_{\mu_j}(n) Y_{z,n} \quad E_z^{\mu_j} R_z^{\mu_j} = \sum_n P_{\mu_j}(n) Y_{z,n} e_{z,n}$$

Let us give an estimate of $Y_{z,0}$:

$$\mu_2 e^{\mu_3} R_z^{\mu_3} - \mu_3 e^{\mu_2} R_z^{\mu_2} = \sum_{n=0}^{\infty} \frac{\mu_2 \mu_3^n - \mu_3 \mu_2^n}{n!} Y_{z,n} =$$

$$Y_{z,0}(\mu_2 - \mu_3) + \sum_{n=2}^{\infty} \frac{\mu_2 \mu_3}{n!} \underbrace{\left(\mu_3^{n-1} - \mu_2^{n-1}\right)}_{<0 \text{ for construction}} Y_{z,n} \leq Y_{z,0}(\mu_2 - \mu_3)$$

So

$$Y_{z,0} \geq \frac{\mu_2 e^{\mu_3} R_z^{\mu_3} - \mu_3 e^{\mu_2} R_z e^{\mu_2}}{\mu_2 - \mu_3} \tag{9.20}$$

Sometimes turn off the laser. We now set a bound also for $Y_{z,1}$:

$$e^{\mu_2} R_z^{\mu_2} - e^{\mu_3} R_z^{\mu_3} = Y_{z,1}(\mu_2 - \mu_3) + \sum_n \frac{1}{n!} Y_{z,n}(\mu_2^n - \mu_3^n)$$

$$\leq Y_{z,1}(\mu_2 - \mu_3) + \frac{\mu_2^2 - \mu_3^2}{\mu_1^2} \sum_{n=2}^{\infty} \frac{\mu_1^n}{n!} Y_{z,n}$$

and in the end

$$Y_{z,1} \geq \frac{\mu_1}{(\mu_2 - \mu_3)(\mu_1 - \mu_2 - \mu_3)} \left(e^{\mu_2} R_z^{\mu_2} - e^{\mu_3} R_z^{\mu_3} - \frac{\mu_2^2 - \mu_3^2}{\mu_1^2} \left(R_z^{\mu_1} e^{\mu_1} - Y_{z,0} \right) \right) \tag{9.21}$$

and the same holds for x. Now, we can find and explicit formula for $e_{x,1}$, $Y_{z,0}$, $Y_{z,1}$, $Y_{x,0}$, $Y_{x,1}$ and, in the end, for r the explicit rate.

1.2 Decoy state in the finite size scenario

In an actual QKD with finite size security our task is to accumulate data up to a certain level and then to analyze the security of a block of data: it is thus better to work directly with the number of counts and to group together all the different decoy levels. Why this particular choice?

As we discussed previously, in the decoy state mechanism we typically separate the different level of decoy and the key rate is determined for each level. This is not

efficient in the finite size analysis, since if we divide our data we are effectively reducing the size of each block while we want the opposite, i.e. to accumulate as much data as possible to reduce the finite size effect (for a large number of bits the finite size security is asymptotically close to the infinite scenario).
How can we group the bits together? We have seen that

$$p_{\mu_j} \longrightarrow \hat{\rho}_{\mu_j} = \sum_n P_\mu(n) \left|n\right\rangle \left\langle n\right|, \tag{9.22}$$

i.e. with a given probability p_{μ_j} we generate a state ρ_{μ_j} with a Possonian distribution $P_\mu(n)$. From the point of view of an eavesdropper, this is equivalent to the state

$$\hat{\rho_A} = \sum_j P_{\mu_j} \hat{\rho}_{\mu_j} = \sum_n \underbrace{\sum_j P_{\mu_j}(n) P_{\mu_j}}_{\tau_n} \left|n\right\rangle \left\langle n\right| \tag{9.23}$$

(with τ_n being the probability of sending an n-photon state), i.e. E sees no difference between the case where A generates a coherent state with mean photon number μ with a given probability p_{μ_j} or the case where she generates an n-photon state with probability τ_n. We can thus see the problem as a **counter-part protocol**, in which Alice sends the state n with probability τ_n and a posteriori she chooses the mean photon number μ with probability

$$p(\mu_j|n) = \frac{P_{\mu_j}}{\tau_n} P_{\mu_j}(n). \tag{9.24}$$

Once again we stress that this is not what is actually done, but is completely equivalent from the point of view of E.

Remembering the conversion from the key rate to the key length we introduced previously (if N is the number of sent pulses and q the sifting probability),

$$\ell_\infty = qN \sum_j P_{\mu_j} r^{\mu_j}. \tag{9.25}$$

We introduce

$$s_{Z,n} = qN \sum_j P_{\mu_j} R_{Z,n}^{\mu_j}, \tag{9.26}$$

a quantity that we can interpret as the number of sifted detection where Alice sends the n-photon state (since we averaged over the different decoy probabilities). We can rewrite it as

$$s_{Z,n} = s_{Z,0} + s_{Z,1} \left[1 - h_2(e_{X,1})\right] - n_Z f_{EC} h_2(E_Z), \tag{9.27}$$

where

$$n_Z = \sum_n s_{Z,n}, \tag{9.28}$$

i.e. the total number of sifted detection in the z-basis, a parameter that we actually measure, while $s_{Z,0}$, $s_{Z,1}$ need to be estimated. This latter task is relatively simple; starting from Eq. (9.27)

$$s_{Z,n} = qN \underbrace{\sum_j P_{\mu_j} P_{\mu_j}(n)}_{\tau_n} Y_{Z,n}, \tag{9.29}$$

where $Y_{Z,n}$ is the **yield**, i.e. the probability that E forwards an n-photon state. Recognizing $\sum_j P_{\mu_j} P_{\mu_j}(n) \equiv \tau_n$ we can write

$$s_{Z,n} = qN\tau_n Y_{Z,n}. \tag{9.30}$$

The advantage of this formulation is that τ_n is known (since it is a parameter of the protocol) and $Y_{Z,n}$ is to be estimated. A way to do it is in terms of the rate; what we actually measure is the parameter

$$n_{b,\mu_j} = qN P_{\mu_j} R_b^{\mu_j}, \tag{9.31}$$

where the terms are (in order) the number of sifted j detection when the μ_j decoy state was sent (which is measurable in the lab), the probability of preparing the coherent state μ_j and finally the rate. We can now define the bound by only using n_{b,μ_j}:

$$s_{Z,0} = qN\tau_0 Y_{Z,0} \geq \frac{\tau_0}{\mu_2 - \mu_3} \left[\mu_2 \frac{e^{\mu_3} n_{Z,\mu_3}}{P_{\mu_3}} - \mu_3 \frac{e^{\mu_2} n_{Z\mu_2}}{P_{\mu_2}} \right] \tag{9.32}$$

since it holds by Eq. (9.31) that

$$\frac{e^{\mu} n_{Z,\mu}}{P_\mu} = Nqe^{\mu} R_b^{\mu}. \tag{9.33}$$

We have also an explicit bound on the (first) error rate

$$e_{X,1} \leq \frac{\tau_1}{(\mu_2 - \mu_3) s_{X,1}^2} \left(\frac{e^{\mu_2} m_{X,\mu_2}}{P_{\mu_2}} - \frac{e^{\mu_3} m_{X,\mu_3}}{P_{\mu_3}} \right) \tag{9.34}$$

where m_{X,μ_2} are the (absolute) total number of error, defined (in a similar way with respect to Eq. (9.31)) as

$$m_{b,\mu_j} = n_{b,\mu_j} E_b^{\mu_j}, \tag{9.35}$$

where $E_b^{\mu_j}$ is the error rate.

In order now to introduce the finite size statistics we need to operate a series of three corrections.
The first step is to understand what the difference between the average value in the finite scenario and the average value in the infinite (asymptotic) scenario. This issue is solved by

Definition 1.1 (Hoeffding's inequality)**.** For n occurrence of a random variable x and in the finite size scenario, the probability

$$p(|\langle x\rangle - \langle x\rangle^{\star}| \leq t|) \geq 1 - 2 \cdot \underbrace{\exp\left(-2t^2 n\right)}_{\equiv\varepsilon} \implies \tag{9.36}$$

$$\left| n_{Z,\mu_j} - n^{\star}_{Z,\mu_j} \right| \leq \delta(n_Z, \varepsilon) = \sqrt{\frac{n_Z}{2} \ln\left(\frac{1}{\varepsilon}\right)} \tag{9.37}$$

where $\langle x\rangle$ is the mean value of x and the $\star$ indicates the true (statistical) value.[2]

This means that the number of photons we measure n_{Z,μ_j} is close (in its absolute value) to the infinite value up to δ. This means in turn that

$$n^{\star} \leq n + \delta \equiv n^{+} \tag{9.38}$$

$$n^{\star} \geq n - \delta \equiv n^{-}. \tag{9.39}$$

By replacing the measured n_{z,μ_j} in Eq. (9.38) with the upper/lower values in Eq. (9.32) (the same goes for Eq. (9.34) with $m^{\pm}_{x,\mu_j}$) we can obtain the **worst case scenario** in the finite size limit and thus the **first finite size correction**.

For the **second** we consider Eq. (9.26): in that relation, we estimate the parameters n_Z, $n_{Z,\mu}$, $m_{Z,\mu}$ via measure on the (actual) data that we will use for the key, while the parameter $e_{X,1}$ is used in a **different set** of data, i.e. the sifted bits in the X basis. While for the infinite scenario the error rate that we measure in X is the same that we measure on Z, for the finite size this is no more the case: the error rate that we measure in X can in general be different to the one we measure on Z for statistics, simply because the two samples could be different in size/length. **Random-sampling theory (without replacement)** comes in help: we correct Eq. (9.26) with a **phase error** ϕ_X as

$$\begin{aligned} s_{Z,n} &= s_{Z,0} + s_{Z,1}\left[1 - h_2(\phi_X)\right] - n_Z f_{EC} h_2(E_Z) \\ \phi_Z &\leq e_{X,1} + \mathcal{O}(\varepsilon, e_{X,1}, s_{X,1}, s_{Z,1}) \end{aligned} \tag{9.40}$$

[2] $n^{\star}_{b,\mu_j} = qNp_{\mu_j}R^{\mu_j}_b$ is what we expect in the asyntotic limit.

with probability $1-\varepsilon$ such that

$$\varepsilon \leq \sqrt{\frac{n_X+n_Z}{e_X(1-e_X)n_X n_Z}}2^{-(n_X+n_Z)\cdot\xi(\theta)} \tag{9.41}$$

$$\xi(\theta) \approx \frac{1}{2}\ln(2)\frac{(1-b_X)b_X}{(1-e_X)e_X}\theta^2 + \mathcal{O}(\theta^3) \tag{9.42}$$

$$b_X = \frac{n_X}{n_X+n_Z}. \tag{9.43}$$

With Eq. (9.40) we have a way to upper bound the phase error and the price we pay is the correction $\mathcal{O}(\varepsilon, e_{X,1}, s_{X,1}, s_{Z,1})$. In the limit n_X, n_Z that go to infinity we retrieve the infinite size scenario.

The last and **third** step involves correction to ℓ due to the smooth entropy. Due to the quantum leftover hashing lemma, we know that we can extract a Δ-secret key of length ℓ, where

$$\Delta \leq \nu + \frac{1}{2}\sqrt{2^{\ell - H^{\nu}_{\min}(Z|E')}} \tag{9.44}$$

and E' is the information of the eavesdropper after all the error correction stage. To clarify, for the min-entropy it holds

$$H^{\nu}(Z|E') \geq H^{\nu}(Z|E) - \lambda_{EC} - \log_2\frac{2}{\varepsilon_{\text{corr}}}, \tag{9.45}$$

where $H^{\nu}(Z|E)$ is the eavesdropper information before error correction: basically from the latter we remove the information that we "reveal" during error correction and verification.
So, if we choose ℓ to be the min-entropy minus a correction term we get a bound on Δ (and thus on the secrecy of the protocol),

$$\ell \equiv H^{\nu}_{\min}(Z|E') - 2\log_2\left(\frac{1}{2\bar{\nu}}\right) \implies \Delta \leq \nu + \bar{\nu}. \tag{9.46}$$

Finally, we get to

$$s_{Z,0} + s_{Z,1}(1-h_2(\phi_X)) - \lambda_{EC} - \log_2\frac{2}{(\alpha_2\alpha_3\tau)^2\varepsilon_{\text{corr}}}, \tag{9.47}$$

where $\varepsilon_{\text{corr}}$ is related to the probability of detecting some remaining errors, while the α parameters can be split in the form

$$\nu = 2\alpha_1 + \alpha_2 + \alpha_3. \tag{9.48}$$

To summarise, finite size entered into the game in three parts: worst case estimation of the decoy parameters, estimation of phase error and finite size correction of the length.

2 Attacks on the implementations

We will now review possible strategies of attack due to non-idealities in the implementations of the protocols. These are divided in attacks on the receiver and on the source.

2.1 Attacks on the receiver (detectors)

Double-clicks. For different causes, it could happen that both detectors fire at the same time (e.g because of dark counts, multi-photon events, ecc.). The most intuitive way to tackle this problem is simply to remove double-click events. However, this introduces a **loophole** in the security. Eve could exploit this fact to manipulate the outcome of the measure. Supposing that we have an active switch capable of choosing to send either H/V or +/- states, if E sends a very strong pulse $|\alpha\rangle$, B will measure either only H/V or will receive a double detection. E is thus capable of deciding in which base B will detect through a intercept and resend attack: for each of the states she measures, she sends a strong laser (with that state) and she knows for sure that all clicks that do not match that very same basis will be thrown away because they generate a double click. The solution is to make a random assignment of the output: this makes the output secure again.
Fake-state attacks. For this class of attacks, E has the ability to control the result of B's outcome. Fake-state alter the detection on Bob's side. We will see blinding, after-gate or laser damage attacks.
To understand them, it is important to first know how an avalanche photo-diode work. Basically, we the electron avalanche is triggered after a characteristic break-down voltage V_{br}, before which the current is negligible. After having reached a certain current I_{th}, a **quenching mechanism** brings the voltage below threshold in order to avoid ruining the detector and re-set it again (which causes some dead (latent) time). Below V_{br} the detector works like any standard linear detector (in a current/power graph, see Fig. 9.1). In this linear regime, the sole intensity of light can be exploited to manipulate a "false" detection: if the current surpasses I_{th} we get a click, otherwise the detector does not see anything. The eavesdropper can send a state, tuning its power such that it has the right amount of energy to

— trigger only one detector if it is measured in the correct basis (which is decided by E);

— distribute the energy equally between both detectors such they do not click if the measurement basis does not coincide with the one chosen.

But how can the eavesdropper access this linear mode? By shining a strong pulse of light, we force the detector from Geiger mode to liner mode, effectively "blinding" it from single photon state (hence the name, blinding attack). On top of this, we superimpose a signal that makes the detector click on demand of the eavesdropper. A

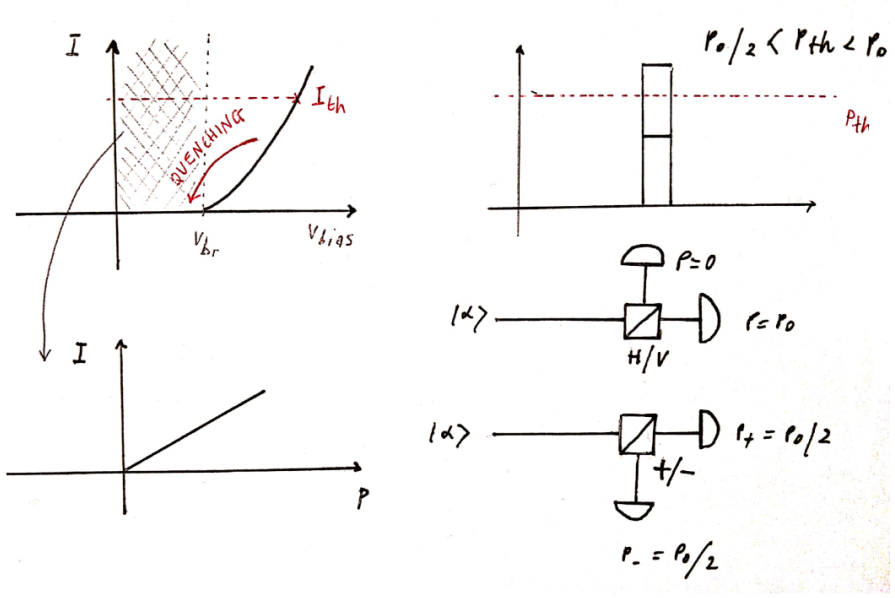


Figure 9.1: Photodiode characteristic curve and blinding attack mechanism.

simple counter measure is to continuously monitor both the current that is produced by the detector and if there is some extra light that is coming to the experimental apparatus.

Some class of detectors are gated, i.e. the bias of the detector is not constant: in the gated detectors we continuously switch above and below the bias voltage (in order to reduce dark counts). In a sense, we turn on the detector only when we expect an incoming photon. This is once again subject to the same procedure seen above: in this case Eve does not even need to shine a strong light since when the detector is turned off it is already in linear mode regime. The solution is again to monitor the power of the incoming light.
Lastly, in the laser damaging attack we change the feature of the detector by means of an even stronger laser light: by increasing the optical power we actively modify the efficiency of the detector, and thus are able to control its properties. The solution is - as before - constant monitoring.
Time-shift attack. Detectors could have different response in time due to the electronics/internal workings (Fig. 9.2). The eavesdropper can at this point manipulate the time of arrival of the incoming photon depending on the characteristic time τ (delaying/anticipating the photon), sending states in the "two tails of the distributions". In this way, E basically decides which state B will see. In an actual system to know τ is delay is non-trivial, also because its value may shift in time.The take-home message is to have detectors as similar as possible in order not to have leaks of security.

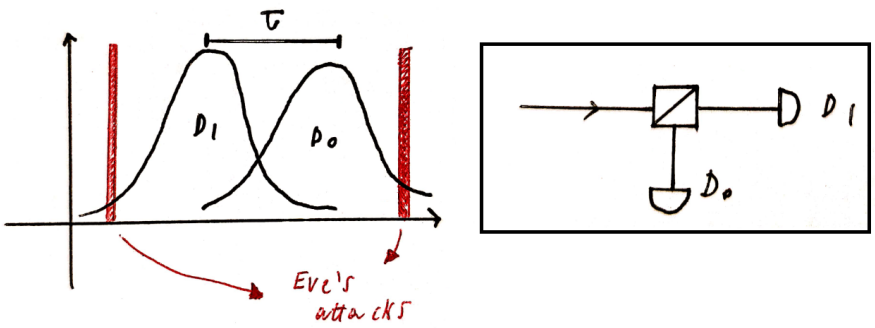


Figure 9.2: Detection efficiencies distributions vs. time for the time-shift attack.

Wavelength-dependent attack. Fiber beamsplitters could have internal wavelength dependence, i.e. may show 50/50 probabilities not for all λs of the incoming light. For certain wavelengths, light could be either totally reflected or totally transmitted. The solution to this weakness is to place filters to block all the light with a different wavelength from the one wanted.

2.2 Attacks on the source

Trojan horse attack. Allows to determine which is the prepared state by Alice by exploiting the reflections of the components: if we have a polarization modulator inside the transmitter, depending on the value of the polarization that we send, if we inject light from outside in to the transmitter we get some light reflected back. The properties of this light depend on the modulator values, i.e. on which state we are preparing. To solve this problem, we place an **attenuator** at the end of the chain, such that if light is injected from outside it is attenuated (twice) before exiting and does not pollute the experiment. Moreover, **optical circulators** could further help our cause: light coming from the inside is propagated normally, while light coming from the outside gets deviated along a different direction.

Unambiguous state discrimination attack (USD). Let's suppose that we do not phase-randomize our decoy states. This kind of attack works for non-orthogonal states: if we try to discriminate between two such states, we succeed with a given probability. The eavesdropper can thus let pass the states which she can precisely determine and blocks all the others. This is more simply understood in terms of qubits: supposing that we want to discriminate between the states $|0\rangle$, $|+\rangle$, we can perform a POVM:

$$\Pi_0 = \gamma |1\rangle\langle 1| \tag{9.49}$$

$$\Pi_1 = \gamma |-\rangle\langle -| \tag{9.50}$$

$$\Pi_2 = \mathbb{1} - \Pi_0 - \Pi_1, \tag{9.51}$$

in the first case for a measure Π_0, we know for sure that if we get an outome, it will be a $|+\rangle$ state; similarly for Π_1 we know that we will get a $|0\rangle$ state. For Π_2 we have no definite answer (no discrimination). This is an example of unambiguous state discrimination, where the discrimination works only with a certain probability.

We've seen how the measurement device is the weakest part of the QKD protocol (since it can do nothing but receive whatever is sent to it). We will now see a class of protocols which are secure against any possible imperfection of the measurement device. The implementation will be more complex but with a gain in security.

3 Full-device independent QKD

Also the full-device independent QKD (like for QRNGs) is based on Bell's inequalities: these state are the way to certify security, since they give us the information of the entanglement state of the two parties.
Let's consider the entangled version of BB84 protocol (where an external source is sending qubits to Alice and Bob), characterized by the correlation terms between the two parties. If a $|\phi^+\rangle$ state is prepared by the source, in the case of perfect communication the correlations are:

$$\left\langle \hat{\mathcal{Z}}_A \otimes \hat{\mathcal{Z}}_B \right\rangle_{\phi^+} = 1 \qquad\qquad \left\langle \hat{\mathcal{X}}_A \otimes \hat{\mathcal{X}}_B \right\rangle_{\phi^+} = 1$$

Where $\hat{\mathcal{Z}} \equiv \hat{\sigma}_z$ (and the same for $\hat{\mathcal{X}}$.
Now, are these max correlations a signal of entanglement or are them achievable with a separable state? Actually, the second statement is correct: consider a four-qubits state, where two of them are sent to each party

$$\rho_{AB} \propto \sum_{z_0=0}^{1} \sum_{z_1=0}^{1} |z_0 z_1\rangle_A \langle z_0 z_1|_A \otimes |z_o z_1\rangle_B \langle z_0 z_1|_B$$

Where $z_0, z_1 \in \{0, 1\}$ indicate the value of the $0-$th and first qubit of each party in the Z basis, respectively. If we now consider measurements for Alice and Bob defined as

$$A_0 = \hat{\mathcal{Z}}_0^A \otimes \mathbb{1}_1^A \qquad\qquad A_1 = \mathbb{1}_0^A \otimes \hat{\mathcal{Z}}_1^A$$

$$B_0 = \hat{\mathcal{Z}}_0^B \otimes \mathbb{1}_1^B \qquad\qquad B_1 = \mathbb{1}_0^B \otimes \hat{\mathcal{Z}}_1^B$$

By measuring $\hat{A}_0$ and $\hat{B}_0$ (or, identically, $\hat{A}_1$ and $\hat{B}_1$) we obtain on the qubit where we apply $\hat{\mathcal{Z}}$ maximal correlations like in the entangled case. To sum up: correlations are an index of entanglement only when the system that we are measuring is fully known, while if we are unsure about the dimension of the system, they can bribe

us (in this example, a larger separable state produces the same correlations of an entangled 2-dim state). Bell inequalities will allow us to quantify entanglement despite of the dimension of the system.
Consider now the case, like BB84, where we move back to having Alice and Bob receiving a single photon each from the source. The measure operators are chosen at random, i.e. Alice draws $x = 0, 1, 2$ and measures $\hat{A}_x$, while Bob draws $y = 1, 2$ and $\hat{B}_y$, collecting outputs $a = \pm 1$ and $b = \pm 1$ (or $0, 1$, just a matter of convention). The measurement operators are now redefined as

$$\hat{A}_0 = \hat{\sigma}_z \qquad\qquad \hat{A}_{1,2} = \frac{\hat{\sigma}_z \pm \hat{\sigma}_x}{\sqrt{2}}$$

$$\hat{B}_1 = \hat{\sigma}_z \qquad\qquad \hat{B}_2 = \hat{\sigma}_x$$

For the key generation, we are interested solely in the case $\hat{A}_0, \hat{B}_1$ (both measure in the Z basis).
The Bell parameter $S_{CHSH} \equiv S$ is used for certification of entanglement - and thus security. It is defined as:

$$\left\langle \hat{A}_1 \otimes \hat{B}_1 \right\rangle + \left\langle \hat{A}_2 \otimes \hat{B}_1 \right\rangle + \left\langle \hat{A}_1 \otimes \hat{B}_2 \right\rangle - \left\langle \hat{A}_2 \otimes \hat{B}_2 \right\rangle = S \tag{9.52}$$

We know want to show what is the key rate here wrt the standard BB84 protocol. We have seen that in the $\infty-$key scenario:

$$r = 1 - \underbrace{f h_2(Q_Z)}_{\text{err corr}} - \underbrace{\chi_E}_{\text{Eve info}}; \tag{9.53}$$

Here the difference lies in the dependence on S of the bound on the eavesdropper information, that in the standard BB84 depends on the error on the other basis (X):

$$r = 1 - f h_2(Q_Z) - \chi_E(S). \tag{9.54}$$

Bell's inequality provide dimension independent entanglement witness!

Proof. Without losing generality, we can picture that any bipartite system A,B characterized by outcomes ± 1 can be modeled as (or reduced to) a qubit-like scenario. Therefore, we can imagine the problem to be projected onto an $(x, z)-$plane of the Block sphere with measurement operators $\hat{A}_x, \hat{B}_y$ acting here:

$$A_X,\, B_Y \in (x, z)\text{-plane}$$

any state that we can generate in here can be reduced to an incoherent superposition of Bell diagonal states in the form

$$\rho_{AB} = \sum p_A \rho_\lambda \qquad\qquad \rho_\lambda = \text{diag}(\lambda_{\phi^+}, \lambda_{\psi^-}, \lambda_{\phi^-}, \lambda_{\psi^+}) = \text{diag}(\vec{\lambda})$$

with $\lambda_{\phi^+} \geq \lambda_{\psi^-}$ and $\lambda_{\phi^1} \geq \lambda_{\psi^+}$. ρ_λ is called a Bell diagonal state (diagonal in the Basis of Bell states). From the BB84 theory we know that the Olevo bound on the info of the eavesdropper is

$$\chi_E = S(\rho_E) - \frac{1}{2}\sum_{b_1=0}^{1} S(\rho_{E|b_1})$$

Where b_1 is the outcome of the measure of $\hat{B}_1$.
We can now consider the whole system, including the eavesdropper, which can be assumed to be pure because of purification:

$$|\Gamma\rangle_{ABE} = \sqrt{\lambda_{\phi^+}}\,\left|\phi^+\right\rangle_{AB}|e_1\rangle_E + \dots$$

Now, consider the most general choice for $\hat{B}_1$ as a measurement operator over the $(x,z)-$plane:

$$\hat{B}_1 = \cos\phi\ \hat{\sigma}_z + \sin\phi\ \hat{\sigma}_x$$

which has eigenvalues (for $b_1 = 0, 1$) given by:

$$|b_1\rangle = \sqrt{\frac{1 + b_1\cos\phi}{2}}\,|0\rangle + \sqrt{\frac{1 - b_1\cos\phi}{2}}\,|1\rangle$$

Then, the state of the eavesdropper depending on b_1 used in the bound above is given by definition by

$$\rho_{E|b_1} = \mathrm{Tr}_A\left[\langle b_1|\Gamma\rangle\,\langle\Gamma|b_1\rangle\right] = \Lambda_+\,|\Lambda_+\rangle\,\langle\Lambda_+| + \Lambda_-\,|\Lambda_-\rangle\,\langle\Lambda_-|$$

Where we stressed its diagonal form with eigenvalues:

$$\Lambda_\pm = \frac{1}{2}\left[1 \pm \sqrt{(\lambda_{\phi^+} - \lambda_{\psi^-})^2 + (\lambda_{\phi^-} - \lambda_{\psi^+})^2 + 2\cos\phi(\lambda_{\phi^+} - \lambda_{\psi^-})(\lambda_{\phi^-} - \lambda_{\psi^+})}\right]$$

In particular, observe how the choice of ϕ in the definition of $\hat{B}_1$ reflects on the form of its eigenvectors and ultimately inside the form of the eigenvalues of Eve's state, together with the ones of the Bell diagonal state. Nevertheless, they do not depend on the value of b_1 (i.e, are the same for $b_1 = 0$ or $b_1 = 1$).
Putting substitution, one can finally see that

$$\frac{1}{2}\sum_b S(\rho_{E|b}) = h_2(\Lambda_+)$$

An in order to consider the worst-case scenario we must maximize χ_E (and therefore minimize $h_2(\Lambda_+)$). One can prove that this is done by setting $\phi = 0 \implies B_1 = \hat{\sigma}_z$ (the one we want to implement in real world. In this case it holds:

$$\chi_E = H(\vec{\lambda}) - h_2(\lambda_{\phi^+} + \lambda_{\phi^-})$$

This is actually the exact same result obtained for the standard BB84 (there, by the means of correlations in the non-key basis). □

The Bell parameter will be our tool to estimate the value of λs above. In particular, let's define the following:

$$R_{\pm} = \sqrt{(\lambda_{\phi^+} - \lambda_{\psi^\pm})^2 + (\lambda_{\phi^-} - \lambda_{\psi^\pm})^2}$$

For a Bell diagonal state it can be shown that

$$S = 2\sqrt{2}\max(R_+, R_-)$$

Following the fact that for a two-qubit state the matrix

$$T_{ij} = \mathrm{Tr}(\sigma_i \otimes \sigma_j \rho_{AB}) \qquad\qquad i,j = 1,2,3$$

such that

$$S = 2\sqrt{\tau_1 + \tau_2}$$

where τ_1, τ_2 are the largest eigenvalues of $T^T T$ (best possible scenario, for optimal legitimate measurement). Following this definition one can finally bound Eve's information:

$$\chi_E \leq h_2\left(\frac{1+\sqrt{(S/2)^2-1}}{2}\right) \text{ for } R_\pm > \frac{1}{\sqrt{2}} \text{ violation of Bell ineq.}$$

What are the differences between this approach and the standard BB84 in a depolarizing channel? In such a channel the state it holds:

$$\begin{aligned}
\rho_{AB} &= p\left|\phi^+\right\rangle\left\langle\phi^+\right| + (1-p)\frac{\mathbb{1}}{4} \\
Q_x &= Q_z = \frac{1-p}{2} \\
S &= 2\sqrt{2}p = 2\sqrt{2}(1-2Q)
\end{aligned}$$

And in therms of rate we find that

$$\begin{aligned}
r_{BB84} &= 1 - 2h_2(Q) \\
r_{DI} &= 1 - h(Q) - h_2\left(\frac{1+\sqrt{4Q^2-8Q+1}}{2}\right)
\end{aligned}$$

As a result, the DI approach is less tolerant, since we are putting less assumptions on the apparatus construction. In fact, in the standard BB84 the minimum QBER for having $r > 0$ is 11%, for DI is 7%. The DI approach works under the assumption of memory-less functioning and time-independent devices, which are non-trivial requirements that impact greatly especially the finite-key scenario. However, to violate Bell inequality we need to close information loopholes. While in std BB84,

losses simply affect the key rate, in here if Alice measures something and Bob does not we need to do something about it, otherwise is impossible to estimate S properly. Critically, the best choice is assigning something to non-detections: assign random outcome to A_0 outcome, and $-1/2$ to all other measures. This allows to stay closer to the boundary of Bell inequality, instead that contributing to the avg S with a 0 (far away from the boundary). In order to close properly the information loophole, an ideal efficiency (above $\eta = 92\%$) should be achieved, which is feasible but hard to extend to long distances.

4 Measurement device independent QKD

DI-QKD is really interesting in theory, but hard to achieve practically. Hence, we picture here a semi-device independent approach for which we require the source to be fully trusted but we allow the measurement device to be under full control of the eavesdropper. The idea is the one of a reverse entanglement generation.
For the sake of simplicity, consider first the one-qubit scenario. Let's suppose that Alice (A) and Bob (B) can generate the states of the BB84 protocol ($|0\rangle, |1\rangle, |+\rangle, |-\rangle$): they send their states to a central station (that cannot be trusted) Charlie (C), which can eventually be Eve (E) under cover. For example, an optical implementation of this is reported in figure, where A and B are sending polarized pulses to encode the states and C mixes the incoming photons by the means of a Beam Splitter (BS), follower by Polarized Beam Splitters (PBSs) and Detectors (Ds) for measurement in the $\{|H\rangle \equiv |0\rangle, |V\rangle \equiv |1\rangle\}$ basis.
DISEGNO
This apparatus is **implementing a Bell state measurement**: in fact, two identical photons arriving at a BS always exit in the same direction (i.e, the "one per side" amplitude probabilities are set to zero via destructive interference, see Hong–Ou–Mandel effect). However, if the two incoming photons are in a joint state $|\psi^-\rangle$ (perfectly anti-correlated), then the exact opposite occurs and we will always have one photon per part. Therefore, we have an experimental way of **post-selecting entanglement**, starting from a separable state: is like our measure is **projecting an initial separable state into an entangled one**: once the two photons arrive, if we have one detection per side we know the state was ψ^-, if we have a double click on either side we know the state was ψ^+.
To give an example, assume Alice and Bob prepare independently their photons such that $|\psi\rangle_A B = |H\rangle_A \otimes |V\rangle_B$. This state can also be written as $|H\rangle_A \otimes |V\rangle_B = \frac{1}{\sqrt{2}}(|\phi^+\rangle + |\phi^-\rangle)$, so by measuring the Bell states $|\psi^\pm\rangle$ we are projecting the separable state onto one of the entangled ones of our measurement basis.
Note that there is instead no way of distinguishing between $|\phi^+\rangle$ and $|\phi^-\rangle$, since both these states (recall $|\psi^\pm\rangle \propto |HH\rangle \pm |VV\rangle$) make just one detector click. Moreover, since all four states of the Bell basis are equally probable for this preparation of Alice and Bob, we have that in the 50% of cases just one detector clicks (meaning

we measured ϕ^+ or ϕ^-, event discarded) while in the other 50% of cases we have a double click ($\psi^\pm$ states). Since we are only interested in the latter case, this protocol discards a lot of photons; nevertheless, we have a theoretical 100% entanglement generation (and consequent security).
Back to the MDI protocol: at this point, Alice and Bob's photons are entangled, and therefore they can proceed with the standard BB84 routine according to what Charlie tells them: after sifting, they look at the measurement outcome and the basis they used to prepare the state and then perform the following: This allows them

	ψ^+	ψ^-
Z	bit flip	bit flip
X	do nothing	bit flip

to solve the anti-correlations and ultimately share the same bit values (basically, in the X basis for ψ^+ the states are perfectly correlated already). It is important to notice that the probability for Charlie to find a $|\psi^+\rangle$ or $|\psi^-\rangle$ state are the same for all possible choices of Alice and Bob states. Moreover, from its output Charlie has no way to tell which states Alice and Bob prepared, due to quantum superpositions. As a result, **the measurement device can be seen as a black box providing a binary output** which gives info to A and B about what to do next.
What if Charlie is indeed malicious? We already stated that if the implementation is correct, the (anti-)correlations are perfect: hence, if Charlie=Eve does not perform a Bell state measurement, it will ruin correlations and enlarge the QBER. Hence, it is sufficient that A and B share a bit of their sifted bits and estimate this quantity to detect the attacker.

The decoy state MDI implementation

So far so good, but in reality we are not really able to prepare single photon states with high fidelity: therefore, we must rely on attenuated pulses and go back to the decoy state technique. Alice and Bob now generate **phase-randomized coherent pulses with given polarization encoding**. In fact, if Alice and Bob can prepare the states $|\alpha_H\rangle, |\alpha_V\rangle, |\alpha_D\rangle, |\alpha_A\rangle$ then phase-randomizing the states we get:

$$\int \frac{d\phi}{2\pi} |\alpha_H\rangle \langle\alpha_H| = e^{-\mu} \sum_n \frac{\mu^n}{n!} |n_H\rangle \langle n_H| \tag{9.55}$$

Which is an incoherent superposition of photon-number states, with Poissonian statistics. This is indeed what we actually send into the protocol.
We can define the rate for the case in which Alice sends n photons, Bob sends m states, in a similar fashion w.r.t. the standard decoy-state:

$$R^{\mu_i,\mu_j}_{Z|n,m} = e^{-\mu_i} e^{-\mu_j} \frac{\mu_i^n \mu_j^m}{n!m!} Y_{Z|n,m} \tag{9.56}$$

When now we have n, m due to the fact that both Alice and Bob are senders. We can generalize also the total error as

$$E_Z^{\mu_i,\mu_j} R_Z^{\mu_i,\mu_j} = R_{Z|n,m}^{\mu_i,\mu_j} e_{Z|n,m} \tag{9.57}$$

Note that for sifted events in the Z basis we have $e^{\star}_{Z|n,m} = 0$, since an error would be the measure of H, H which is automatically discarded (since it belongs to the $\phi^{\pm}$ states). In the X basis, this is not true (in practice, $E_x^{\mu_i,\mu_j} \neq 0$). Anyway, for the single-photon cases it holds $e^{\star}_{X|1,1} = 0$.
It is important to mention that actually the Hong–Ou–Mandel effect exploited for the Bell state measurement is perfect for the two-photons case but not perfect for the case of phase-randomized attenuated pulses (which include multi-photon scenarios). Therefore, errors can in practice be higher than zero, even if still not very large.
With the generalizations above for the two-senders decoy the key rate translates into the following:

$$r^{\mu_i\mu_j} = R_{Z|1,1}^{\mu_i,\mu_j}(1 - h_2(e_{X|1,1})) - R_Z^{\mu_i,\mu_j} h_2(E_Z^{\mu_i,\mu_j}) \tag{9.58}$$

Where we neglect the zero-photon component. One can analytically find a bound to the rate by the means of Eq. (9.56) and Eq. (9.57). For $\mu_1 > \mu_2 > \mu_3 = 0$

$$R_Z^{(2)} = R_Z^{\mu_2,\mu_2} e^{2\mu_2} + R_Z^{0,0} - e^{\mu_2}(R_Z^{\mu_2,0} + R_Z^{0,\mu_2}) \tag{9.59}$$

$$R_Z^{(1)} = (\mu_2 \longleftrightarrow \mu_1) \tag{9.60}$$

$$\tag{9.61}$$

Which leads to a bound on the yields (i.e, probability to arrive at measure):

$$Y_{Z|1,1} \geq \frac{\mu_1^3 R_Z^{(2)} - \mu_2^3 R_Z^{(1)}}{\mu_1^2 \mu_2^2 (\mu_1 - \mu_2)} \underset{(a)}{\approx} \eta^2 \frac{\mu_1 e^{2\mu_2} - \mu_2 e^{2\mu_1}}{\mu_1 - \mu_2} \tag{9.62}$$

where in (a) we exploited that for a lossy channel: $Y_{Z|1,1} = \eta^2$. Notice that $R_Z^{(1)}$, $R_Z^{(2)}$ and $E_X^{\mu_i,\mu_j}$ are estimable experimentally. Problem: the two photon have to be **identical** (indistinguishable) and specifically at the same wavelength, since otherwise they do not interfere at the BS surface; also: synchronization issue much more stringent than in the BB84 protocol, which was only limited to the precision of the detector (ns accuracy/sync). This can be done via phase-locking of lasers or exploiting atomic clocks. Overall, we gain in terms of security but we increase the complexity.

Chapter 10

Twin-field QKD

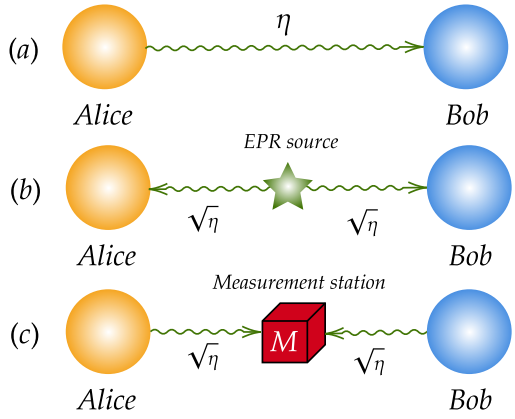


Figure 10.1: Three cases of QKD up to now: in (a) a prepare and measure architecture (PM), in (b) a source-device independent (SDI) one and in (c) a measurement-device independent (MDI).

Let us now refer to Fig. 10.1. In a PM scenario Alice sends a photon to Bob. The distance between the two is a **physical distance**, a communication channel with an efficiency η (that is the probability that one photon arrives at the receiver). The key rate scales with $r \propto \eta$.
In the SDI case, where we have an entangled state produced by a source placed in the middle, the two photons travel along the two channels. Each of them has an efficiency of $\sqrt{\eta}$, and in order to measure an entangled pair we need that two photons are detected. Again, also in this case the rate scales with $r \propto \eta$.
The third case is the MDI, where Alice and Bob generate the photons and the central station performs the Bell state measurement. For some similar considerations, also here $r \propto \eta$. We are not changing the security, but only the security assumptions. The idea behind Twin Field QKD (TF) is to overcome this scaling and reach $r \propto \sqrt{\eta}$!
The very crucial idea behind this approach is to **start from an MPI** approach,

and instead of post-selecting two-photon entanglement, we do it on **single-photon entangled states**. Even with a good grounding in quantum mechanics, one can legitimately wonder what a single-photon entanglement is. We are now going to explain this, but before we recall that with only one photon in an MPI we may reduce to an efficiency that scales as $r \propto \sqrt{\eta}$! We gain a **quadratic factor**.

1 Single-photon entanglement

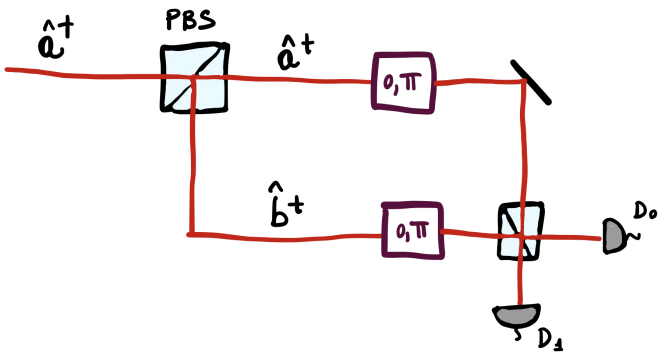


Figure 10.2: A single photon entanglement is created after the first PBS, then Alice and Bob insert a phase and the beams are recombined in a beam splitter and two PDs click accordingly.

We refer to the drawing in Fig. 10.2. Let's suppose we have a single photon that goes towards a PBS. Out, we have a superposition of two modes. The quantum state become

$$\hat{a}^\dagger |0\rangle \to \frac{1}{\sqrt{2}}(\hat{a}^\dagger + \hat{b}^\dagger)|0\rangle \overset{(a)}{=} \frac{1}{\sqrt{2}}(|1\rangle_a |0\rangle_b + |0\rangle_a |1\rangle_b)$$

A single photon is splitted in two optical modes. The rewriting in (a) is much clearer, we are doing a superposition of one photon in a and of the vice versa. This is **single photon entanglement**, between vacuum and a single photon. Let's now suppose that this entangled state is shared between Alice and Bob, and that they can apply a phase $[0, \pi]$ to the state. After this, the two beams are **recombined** in a BS, and **depending on the phase** applied by the two parties, we can have different outputs from the beam splitter. If the two phases are both 0 or π, we have D_0 ($|\psi^+\rangle$) at the output, while if they are $0 - \pi$ or $\pi - 0$ we have D_1 ($|\psi^-\rangle$) instead. For the latter, we can say that applying a π phase in one of the two arms means flipping sign.
Notice that the clicks in the two detectors (used to create a key in the end) tell only if A and B are **correlated** or **anti-correlated**, there is now way to know which of

the two applied a specific phase. If prepare with unit efficiency the photons before Alice and Bob, then we need that just one of them arrives at the end, and so the total efficiency is now only $\sqrt{\eta}$. What we have seen is the general, a bit abstract and surely hard to implement (it is hard to generate a perfect single photon) idea of the TF QKD. However, the idea of twin field is to **remove the part** before Alice and Bob, and generate from them something that resembles the original situation. This idea was proposed in 2018 and refined after few months with the **sending or not sending** protocol (SNS) ([Wang *et al.*(2018)Wang, Yu, and Hu] for the theory, [Hu *et al.*(2019)Hu, Jiang, Yu, and Wang] for the practical implementation). Both the original TF and SNS-TF QKD are MPI protocols with single photon entanglement. *How the twin field really works?* The idea is that Alice and Bob have two possibilities: whenever they **choose to send a decoy** (also called X-basis), they send a state given such as

$$\left|\sqrt{\nu_j}e^{i\delta_A}\right\rangle, \left|\sqrt{\nu_j}e^{i\delta_B}\right\rangle \quad \nu_j = \nu_1, \ldots, \nu_n$$

where ν is the intensity (few values) and δ a completely random phase. Alice and Bob have a second possibility: they can choose to **send a signal state** (also called Z-basis):

$$\varepsilon \to \left|\sqrt{\mu}e^{i\delta_A}\right\rangle \quad 1-\varepsilon \to |0\rangle$$

This gives an idea of why this protocol is called SNS. They choose **randomly** to send decoy or signal. Let us now analyze what happens in the central station: Charlie post select **only single photon** detection, 2 photons or 0 photon are not considered as a successful event. One photon, the ideal case, means also that Alice sends something and Bob sends nothing (or viceversa). Actually we need to do a better estimation: the signal state that is generated, considering that the phases are random and not communicated to anyone, is a **incoherent state** (incoherent superposition of number states)

$$\begin{aligned}\rho_{AB} &= \left[(1-\varepsilon)\left|0\right\rangle_A\left\langle 0\right|_A + \varepsilon\sum_n \frac{e^{-\mu}\mu^n}{n!}\right] \otimes \left[(1-\varepsilon)\left|0\right\rangle_B\left\langle 0\right|_B + \varepsilon\sum_n \frac{e^{-\mu}\mu^n}{n!}\right] \\ &= \sum_n P_n^Z(\mu)\left|\psi_n^Z\right\rangle\left\langle\psi_n^Z\right|\end{aligned}$$

Clearly, the first terms are

$$\begin{aligned}P_0^Z(\mu) &= (1-\varepsilon+\varepsilon e^{-\mu})^2 \\ P_1^Z(\mu) &= 2\varepsilon\mu e^{-\mu}\sqrt{P_0^Z}\end{aligned}$$

When we prepare some state with given probability we can think that with some probability we are preparing the vacuum on both sides and with some probability

one photon here or one photon there:

$$\left|\psi_0^Z\right\rangle = |00\rangle \quad \left|\psi_1^Z\right\rangle\left\langle\psi_1^Z\right| = \frac{1}{2}(|01\rangle\langle 01| + |10\rangle\langle 10|)$$

The parameter ε should be low, to avoid the unwanted cases in which both A and B prepare a photon state and a single photon is measured.
Like in a standard PM, after the run of the protocol, Alice and Bob communicate which are signals and which are decoy states. Only for the decoy state they communicate the precise value of the decoy chosen and the one of the phase. This is why for the signal state we have a completely incoherent superposition, while for the decoy this is not the case because the phase is random but known to the parties. Furthermore, we post-select the data in which the phase is

$$1 - |\cos(\delta_A - \delta_B)| \leq \lambda \tag{10.1}$$

So when the **difference is small** it means that we are keeping the data in which we have

δ_A	δ_B
0	0
0	π
π	0
π	π

Table 1: All the phase possibilities in which $1 - |\cos(\delta_A - \delta_B)| \leq \lambda$. In this case $\delta_- \approx 0, \pi$ is fixed.

But looking at the table, we notice that these are just the cases that matter to us: we are exactly preparing the single-photon entangled states. Let us see this under another point of view. Defining $\delta_\pm = \delta_A \pm \delta_B$, and selecting data for which Eq. (10.1) holds it means that we are fixing the δ_- (the difference), while δ_+ is still undefined. The state in the X-basis (decoy) can be defined as

$$\rho_{AB}^X = \int \frac{d\delta_+}{2\pi} \left|\sqrt{\nu}e^{i\delta_A}\right\rangle\left\langle\sqrt{\nu}e^{i\delta_A}\right| \otimes \left|\sqrt{\nu}e^{i\delta_B}\right\rangle\left\langle\sqrt{\nu}e^{i\delta_B}\right|$$

where we have the same decoy ν, and since δ_- is fixed the integration runs only on δ_+. Without post-selection we should integrate also on δ_- and obtain a state in the Z-basis (incoherent superposition). This is an easy integral because we can rewrite it in a discrete form:

$$\sum P_n^X(\nu) \left|\psi_n^X(\delta_-)\right\rangle\left\langle\psi_n^X(\delta_-)\right|$$

again where we have different contributions with different number of photons. Explicitly,

$$P_0^X(\nu) = e^{-2\nu} \quad \rightarrow \quad |\psi_0\rangle = |00\rangle$$
$$P_1^X(\nu) = 2\nu e^{-2\nu} \quad \rightarrow \quad |\psi_1\rangle = \frac{1}{\sqrt{2}}(|01\rangle + e^{i\delta_-}|10\rangle)$$

The important part is the one photon component: this equation tells us that **doing this post-selection**, we probabilistically prepare the required single-photon entanglement. With a measurement on the latter, we are able to **quantify the phase error**, and so the information related to the eavesdropper.
The formula for the rate is

$$r = P_1^Z(\mu)Y_1[1 - h_2(e_X)] - R_z f_{EC} h_2(E_Z) \tag{10.2}$$

This equation is similar to the one for the PM and measure case, apart from the fact that the first probability is not the Poissonian, and the other term in the first multiplication must be treated differently.
Experimentally, to perform this we need first order interference, i.e. interferometric stability between two paths (a giant Mach-Zender interferometer). One need to preserve the precise relation between the two phases δ_A and δ_B (**phase locking** between two distant places, common phase relation between the two lasers). For MPI we need frequency locking, here we need phase locking (and this is much more difficult to achieve).
Finally, the parameter λ is fixed with a **trade-off**: small lambda means few errors in the phase, but big losses in the post selection phase.

Chapter 11

Quantum repeater

In this chapter, the aim is to have an entangled state between two distant parties Alice and Bob. This could be achieved by splitting the distance into **sub-levels**. DISEGNO The idea is that we can extend, via **entanglement swapping** on the connections, the entanglement created in the elementary parts of the total distance. We create entanglement in L if we have it in $L_0 = L/2^n$, with n equal to the number of levels. We can split our discussion in two main parts, entanglement creation and swapping.

1 Entanglement generation

The first building block is the generation of entanglement, and we have to do it in a **heralded way** (need to know when there is entanglement). One thing that must be taken into account is that in this part we need to figure out also how to keep entanglement alive up to the moment that we are sure that it has been propagated to other sub-levels. To do this, we can exploit a primordial **quantum memory**, following the Duan-Lukin-Cirac-Zoller (DLCZ) protocol [Duan *et al.*(2001)Duan, Lukin, Cirac, and Zoller].
The basic element of our system is a cloud of N_a identical atoms. A pair of meta stable lower states $|g\rangle$ and $|s\rangle$ can correspond e.g. to Zeeman sub-levels of electronic ground state of alkali atoms. All the atoms are **initially prepared** in the ground state $|g\rangle$. We describe the state of the system as:

$$|0\rangle_s = \frac{1}{\sqrt{N_a}} \bigotimes_{i=1}^{N_a} |g\rangle_i$$

A sample is illuminated by a short, **off-resonant** laser pulse that induces Raman transitions into the states $|s\rangle$. Raman scattering is conceptualized as involving a virtual electronic energy level which corresponds to the energy of the slightly detuned exciting laser photons. The absorption of a photon excites the molecule

to the imaginary state and re-emission leads to Raman Stokes, Raman anti-Stokes (both inelastic) or Rayleigh scattering (elastic). The main difference between these relies in the **energy of the emitted photon**. At this point, we are interested in the **Raman Stokes** inelastic scattering, where the photon has an energy proportional (up to h) to the difference between the frequency of the virtual state and the meta stable state higher in energy $|s\rangle$. We are not going to describe this further as this would go beyond the scope of this discussion, but we must point out that this process, since we are off-resonance, happens with very small probability and we can say that **only one atom** from the cloud undergoes this process. To recap: we have a cloud in $|g\rangle$, we shine some light and only one atom goes from $|g\rangle$ to $|s\rangle$. This is the 0 mode:

$$|1\rangle_s = \frac{1}{\sqrt{N_a}} \sum_{k=1}^{N_a} e^{i(\vec{k}_w - \vec{k}_s)x_k} |g\rangle \dots |s\rangle_k \dots |g\rangle_{N_a} = S^\dagger |0\rangle_s \tag{11.1}$$

and since we don't know which is the atom that jumps, we write a superposition of all the possibilities we have with the given cloud. The phase in front the separable state, take into account the wave vectors of the WRITE pulse k_w (we are going to call the detuned laser pulse as **WRITE pulse**), of the Stokes photon k_s and of the position of the hopping atom x_k. We are storing some excitation. We can retrieve this excitation by sending a **READ pulse**. To read the signal we send a laser that is **on-resonance** with $|s\rangle$ and $|e\rangle$, and another inelastic process, the Raman anti-Stokes, happens. Here the anti-Stokes photon has a frequency equal to the difference between the of the first excited $|e\rangle$ and the one of the ground state $|g\rangle$ (greater than the one of the exciting pulse). After the read, I have the following state:

$$|1\rangle = \frac{1}{\sqrt{N_a}} \sum_{k=1}^{N_a} e^{i(\vec{k}_w - \vec{k}_s)x_k} e^{i\vec{k}_r \vec{x}'_k} |g\rangle \dots |e\rangle_k \dots |g\rangle_{N_a} \tag{11.2}$$

where $\vec{x}'_k$ is the new position of the atom. So one atom is in the excited state and will decay to $|g\rangle$ emitting an anti-Stokes photon. The output probability can be written as

$$\frac{1}{N_a} \sum_{k=1}^{N_a} e^{i(\vec{k}_w - \vec{k}_s)x_l} e^{i(\vec{k}_r - \vec{k}_{as})x_l} \tag{11.3}$$

In the writing pulse we have a **collective excitation**, i.e. a superposition of excitations. In the reading process, all the terms can decay on the ground state again. If the phases are all aligned, we have constructive interference and this becomes a very efficient process. So, if $\vec{k}_w - \vec{k}_s + \vec{k}_r - \vec{k}_{as} = 0$ (phase-matching) the reading process is very efficient and we can retrieve the state of the initial photon. This is a quantum memory because we are storing the excitation of the EM field (a photon) in an atom. In fact, all the writing process is at a single-photon level: to excite one atom from $|g\rangle$ to $|s\rangle$ we need only one photon. Physically, the quantum state after

the WRITE pulse can spontaneously decay in the ground state, and the lifetime τ_s of this state is exactly what characterize the quantum memory. The information of the single photon will survive in a time window $[0, \tau_s]$. DISEGNO We now have two clouds of atoms in $|g\rangle$, we send them a WRITE pulse so that, when we see the Stokes photon, we know we are generating

$$\left[1 + \sqrt{\frac{p}{2}}(\hat{a}^\dagger \hat{s}_a^\dagger + e^{i\phi}\hat{b}^\dagger \hat{s}_b^\dagger) + O(p)\right]|0\rangle \tag{11.4}$$

The process happens with small probability $\sqrt{p/2}$ and we neglect multiple excitations. We are basically creating one photon in the cloud A or one in the cloud B, and when we detect a single-photon events at our detectors, we erase the information and we create a superposition between excitation in A or in B

$$|\psi\rangle = \frac{1}{\sqrt{2}}(\hat{s}_a^\dagger + \hat{s}_b^\dagger)|0\rangle = \frac{1}{\sqrt{2}}(|1\rangle_a |0\rangle_b + |0\rangle_a |1\rangle_b)$$

where the operator $\hat{s}^\dagger$ are transforming the ground state in the one that describes the collective excitation (we have one state in $|s\rangle$ but we do not know which one). The state that we are generating is $s|0\rangle = |1\rangle$. We have one excitation, but we do not know if it is on the first or on the second cloud. This process creates an entanglement between the cloud and the photon, so this is entanglement swapping. This is not a deterministic preparation of entanglement: this is heralded, i.e. when we see a photon we know that the spooky action at a distance is present. After this process $|\psi^+_{a,b_1}\rangle |\psi^+_{b_2,c}\rangle$.

2 Entanglement swapping

To create entanglement between adjacent nodes, we perform a READ process, so that we can read the excitation. This process is very efficiently transforming the atomic excitation in a anti-Stokes photon excitation:

$$\underbrace{|1\rangle_S}_{\text{atom}} \to \underbrace{|1\rangle_a}_{\text{photon}} \tag{11.5}$$

In fact, after the heralding we have

$$|\psi^+_{ab_1}\rangle |\psi^+_{b_2c}\rangle = \frac{1}{2}(\hat{s}_a^\dagger + \hat{s}_{b_1}^\dagger)(\hat{s}_{b_2}^\dagger + \hat{s}_c^\dagger)|0\rangle$$

that, after the READ pulse is sent, and by calling as $b_1^\dagger$ and $b_2^\dagger$ the photon excitations, becomes

$$\frac{1}{2}(\hat{s}_a^\dagger + b_1^\dagger)(b_2^\dagger + \hat{s}_c^\dagger)|0\rangle$$

After this, we place a BS and perform a single-photon entanglement post-selection. Hence, we keep the single photon component

$$\frac{1}{2}(b_1^\dagger \hat{s}_c^\dagger + b_2^\dagger \hat{s}_a^\dagger)$$

and by detecting after the BS we have

$$\frac{1}{\sqrt{2}}(\hat{s}_c^\dagger + \hat{s}_a^\dagger)\,|0\rangle$$

To recap: we create the zero level entanglement by a inefficient process of writing a single photon excitation to an atomic cloud. This happens with low probability as already said but we use it as a **trigger** for entanglement generation. The READ process is used to erase the information of where is the collective excitation coming from in terms of atomic clouds. The two in the end where not both in the ground or both in the excited state and by doing post-selection between these two B_1 B_2 we are actually creating entanglement between A and C. The excitation must be kept inside the quantum memory until the moment in which entanglement is established in the other part too. If we have an excitation between $A - B_1$, we need to wait also for an excitation between $B_2 - C$ to perform entanglement swapping: **keep entanglement alive**! Research is making many efforts to increase the efficiency of these memories.

Chapter 12

Alternatives to QKD

We now move away from QKD: we will be looking at other cryptographic mechanisms, on one hand to see some alternatives and on the other to see how they are not as effective as QKD. We will analyze two examples.

1 Quantum information commitment

> Every great magic trick consists of three parts or acts. The first part is called "The Pledge". The magician shows you something ordinary: a deck of cards, a bird or a man. He shows you this object. Perhaps he asks you to inspect it to see if it is indeed real, unaltered, normal. But of course... it probably isn't. The second act is called "The Turn". The magician takes the ordinary something and makes it do something extraordinary. Now you're looking for the secret... but you won't find it, because of course you're not really looking. You don't really want to know. You want to be fooled. But you wouldn't clap yet. Because making something disappear isn't enough; you have to bring it back. That's why every magic trick has a third act, the hardest part, the part we call "The Prestige".
>
> *Cutter, The Prestige*

1.1 Information commitment

The goal of two-party cryptography is to enable two parties, Alice and Bob, to solve a task in cooperation even if they do not trust each other. An example of such a task is the cryptographic primitive known as bit commitment. A bit commitment protocol traditionally consists of two phases: In the commit phase, Bob (B) commits a bit to Alice (A), who receives some form of confirmation that a commitment has been made. In the open phase, Bob reveals the bit to Alice. Security means that

Bob should not be able to reveal anything but the committed bit, but nevertheless Alice cannot gain any information about the bit before the open phase.

Let's put it in formal terms: A wants to prove to B he knows $X \in \mathcal{X}$.

— **Commit**. The "envelope" is in the information theory case just another piece of information we call Y, related to X. A computes Y from X ($A \to B$: Y).

— **Open the commitment**. A sends X to B ($A \to B$: X) and B checks the correspondence $X \leftrightarrow Y$.

Possible applications of information commitment include two party gates (like in the case of the magic trick), delayed authentication. To explain the latter, we have to think of a broadcast context in which the transmitter signs with secret key, the same that is used for verification. Of course, if he discloses the secret key before transmitting, all communications would be compromised; the solution is to wait until we are sure that everybody got the intended message and then disclose the key: in this way, nobody got the secret key in time to forge the signature. However, how can we be sure that the key is authentic? We commit the value of the key, put it into the message and sign it together with the message: in this way we can verify that indeed the key is secure.

For a commitment protocol we require the following:

— **Binding property**. A cannot send $X' \neq X$ without B detecting it;

— **Concealment**. B cannot learn[1] X from Y.

The **classicaly computational secure** solution is to use **cryptographic hash function**, which is not universal hashing (the latter is a random since we choose randomly which $f \in \mathcal{F}$ is used, while the former is deterministic) and has the following properties:

— **Pre-image resistance**. Once we know the result of applying h to X, $Y = (X)$, it is not computationally feasible to learn the input X: it is a one way function, i.e. it easy to compute but hard to invert (any algorithm returns X with very low probability or requires too much resources). This guarantees computational secure concealment.

— **Collision resistance**. It is not feasible to find 2 inputs X_1 and X_2 that will match the same output Y. This guarantees the binding property.

[1] In this case, "to learn" is not intended in the deterministic sense, but rather in the sense that it is unlikely we get the correct answer with a probabilistic algorithm.

Why don't we replace cryptographic hashing with universal hashing function and gain unconditional security? If the value of the particular function is exposed during commitment, we lose the concealment property, since once we choose the key, the function is invertible (since they are linear). On the other hand, if we disclose it during the opening, for A it is still possible to find two elements that will give the same output, and we lose the binding property.

1.2 Early quantum proposals

The first proposal of a quantum commitment scheme comes in the BB84 paper. In the protocol, the information is a single bit $x \in \{0,1\} \equiv B$; we consider the BB84 basis (we will see the meaning of sub- and super-scripts in a moment):

$$\begin{array}{cccc} \updownarrow & \longleftrightarrow & \swarrow\!\!\nearrow & \nwarrow\!\!\searrow \\ \gamma_1^0 & \gamma_0^0 & \gamma_0^1 & \gamma_1^1 \end{array} \tag{12.1}$$

— **Commit**. A will generate a random sequence $\vec{C} \in \mathbb{B}^n$ and will encode every bit $c_i \to \left|\gamma_{c_i}^x\right\rangle$, i.e. he commits the bit by choosing the basis and then for each of this $\vec{C}$ he chooses the state: if he wants to commit to 0 (1) he chooses the linear (diagonal) basis. A sends to B the quantum state, $A \to B$: $|\gamma\rangle \equiv \bigotimes_i \left|\gamma_{c_i}^x\right\rangle$. B measures $|\gamma\rangle$ with random basis $D_i \in \mathbb{B}$: the result is a sequence of binary variables we will call c_i'.

— **Open**. A sends to B both the commit message and all the bits he uses to encode ($\vec{C}$), $A \to B$: x, $\vec{C}$: B will have to check that $c_i' = c_i$, $\forall i$ $D_i = x$, i.e. for an equal basis choice B gets the same result as A.

Why does this protocol work? Intuitively, the density matrix of the γ state is the identity in both cases, whether $x = 0$, 1. Whatever measurement B will try to apply, she will be never able to distinguish between the two. In detail, for the requested properties:

— **Concealment**. $\rho = |\gamma^x\rangle \langle\gamma^x|$ is independent of x.

— **Binding**. If A opens $\bar{x}$ (the opposite value for x), $P[c_i' = c_i] = \frac{1}{2}$, $\forall i$ $D_i = \bar{x}$. The probability of cheating in this case is $\left(\frac{3}{4}\right)^n$.

After some years different variants of this protocol, aimed at improving the information commitment, two papers in the same issue of the same journal proved that it was impossible to reach it. We will show how in the next section.

1.3 Impossibility result

The claim of quantum cryptography has always been that it can provide protocols that are unconditionally secure, that is, for which the security does not depend on any restriction on the time, space, or technology available to the cheaters. [Lo and Chau(1997)] and [Mayers(1997)] showed independently that this claim does not hold for any quantum bit commitment protocol. Since many cryptographic tasks use bit commitment as a basic primitive, this result implies a severe setback for quantum cryptography.

We will now show the basic idea under [Lo and Chau(1997)]. Let $\rho_{AB} = |\psi_{AB}\rangle \langle\psi_{AB}|$ be the state after the commit (we consider pure states without loss of generality): ideally B has no info on the value committed x from ρ_B (i.e. what he observes of ρ_{AB}, $\rho_B = \mathrm{Tr}_A\, \rho_{AB}$).
This means that

$$\rho_B^0 = \rho_B^1 \implies \mathrm{Tr}_A \left[\left|\psi_{AB}^0\right\rangle \left\langle\psi_{AB}^0\right| \right] = \mathrm{Tr}_A \left[\left|\psi_{AB}^1\right\rangle \left\langle\psi_{AB}^1\right| \right]. \tag{12.2}$$

This holds by the so called **unitary equivalence of purifications**, which states that $\exists\, U_A$ unitary transformation such that

$$(U_A \otimes \mathbb{1}_B) \left|\psi_{AB}^0\right\rangle = \left|\psi_{AB}^1\right\rangle \tag{12.3}$$

i.e. there is a unitary transformation on A that she can apply on its side to change the result after the commitment: in other words, A can always cheat (we lose the binding property).
Less ideally from the previous case, B has little information on x from ρ_B, meaning that

$$d(\rho_B^0, \rho_B^1) = \frac{1}{2} \mathrm{Tr} \left|\rho_B^0 - \rho_B^1\right| = \varepsilon > 0. \tag{12.4}$$

The **fidelity** for this distance is at least

$$F(\rho_B^0, \rho_B^1) \equiv \max_{\psi'} \left|\left\langle \psi'^{\,0}_{AB} \middle| \psi'^{\,1}_{AB} \right\rangle\right| \geq (1-\varepsilon)^2, \tag{12.5}$$

where the maximum is computed over the purifications of one fixed density matrix while varying the other. A can operate on his side by changing $\psi'^1_{AB} = (U_A \otimes \mathbb{1}_B)\psi^1_{AB}$, since both ψ'^1_{AB}, ψ^1_{AB} have the same ρ_B^1. Because of the fidelity property in Eq. (12.5), in the opening phase, since the states $\psi'^{\,0}_{AB}$, $\psi'^{\,1}_{AB}$ are nearly aligned, they will be nearly indistinguishable. We have a trade-off: good concealment (i.e. small ε) implies poor binding and vice-versa.

So where is the flow in the protocol?
The protocol can be indeed violated by using entanglement: in the commit phase, A does not really commit and uses Bell states (for instance a pair of entangled

photons), keeps one and sends the other to B, in whatever polarization she likes. At the opening of x, she measures (right before the opening phase) with $\vec{D} = \bar{x}$, and announces c_i. The result will be $c'_i = c_i$, $\forall i$ $D_i = \bar{x}$ and the protocol is cheated.
This trick holds in the widest possible assumptions: with some stronger assumptions, further proposals arise.

1.4 More recent proposals

In the most general assumptions, the impossibility result holds and we have no way to information commitment: stronger assumptions are needed. We will review some of the latest approaches.

Quantum memories, proposed in [Konig *et al.*(2012)Konig, Wehner, and Wullschleger]. Quantum memories can be made non-ideal in in A and/or B. Since the amount of noise may of course depend on the storage time, the behavior of the storage is completely described by the family of maps $\{\mathcal{F}_t\}_{t>0}$, a completely positive trace-preserving map (CPTPM), with the minimal assumption that the noise is Markovian[2], that is the family $\{\mathcal{F}_t\}_{t>0}$ satisfies $\mathcal{F}_{t_1+t_2} = \mathcal{F}_{t_1} \circ \mathcal{F}_{t_2}$.
The idea behind it is that this introduces certain time delays Δt which force any adversary to use his storage device for a time at least Δt. This assumptions imply that the best an adversary can do is to read out the information from the device immediately after time Δt, as any further delay will only degrade his information further.
This protocol makes use of **weak string erasure** of which more details can be found in the related paper.

Relativistic constraints, proposed in [Kaniewski *et al.*(2013)Kaniewski, Tomamichel, Hanggi, and Wehner]. The idea is not to separate A and B but rather to split A (or B) $A \to (A, A')$ during both the waiting and opening phase. More precisely, one imagines that each party is split up into multiple agents who cannot communicate with each other for at least some parts of the protocol. Intuitively, the use of non-communicating agents can evade the standard no-go argument because while all agents in total have enough information to cheat, no single agent can cheat on his own.
In detail:

— A, A' observe x;

— $A \to B$: $y = x$; $A' \to B$: $y' = x$. If both A and A' are honest, B receives the same value and this signals a correct opening. The two parties A and A'

[2] As a side note the noise in storage only increases with time (i.e. the fidelity only degrades through time): this property is essential to ensure that the adversary cannot gain any information by delaying the readout.

cannot agree on a different value of x after the waiting phase because of the distance they cannot effectively communicate.

The same can be said for B (verifier split); $B \to (B, B')$ and

— A generates $z \in \mathbb{B}$ and $A \to B : \bar{x} \oplus z$, $A \to B' : z$;

— Compute $x = (x \oplus z) \oplus z$.

Random oracle, proposed in [Gama *et al.*(2020)Gama, Mateus, and Souto]. Let's consider $x \in \mathbb{B}^{2n}$ where n is the number of entangled pairs; A generates random basis D_i and measures z_i. The commitment value will be

$$y = (x \oplus \mathcal{RO}(\vec{D}, \vec{z}),\ \mathcal{RO}(\vec{D})) \tag{12.6}$$

where $\mathcal{RO}$ is a **random oracle function** that gives the same output for the same input and uniform, independent outcomes for different inputs (we ignore how this could be implemented experimentally). y is sent to B, $A \to B : y$.
For the opening phase, $A \to B : \vec{D}$, i.e. x will remain private: B checks if $\mathcal{RO}(\vec{D}) = y_2$[3] (measuring $D_i \to z_i$). If that is the case, she will compute $x = y_1 \oplus \mathcal{RO}(\vec{D}, \vec{z})$.

2 Digital signature

Digital signatures are mechanisms that provide the following services:

— Message source authentication (the receiver can be sure that the source of the message is trusted);

— Integrity protection (the message is received exactly in the way it was sent);

— 3rd party attestation (sender non repudiation) (the sender cannot repudiate the incoming message).

The first two can also be provided in a classical framework by the means of unconditional security mechanisms (e.g., using common universal hash functions). However, the digital signature in classical communication is two folded: this is efficient! Imagine in the case of one sender and many receivers: if we want to provide authentication and integrity protection by the means of symmetric keys, the sender would need one pair of keys for each user (that have to be distributed, and this might be an issue for large numbers). Therefore, it would be great to have asymmetric encoding.
There is another reason: having a document signed by a person "X" allows me to prove to **someone else** ("Y") the integrity of the document: it comes from the signer if it is signed (e.g., a contract: go to court and say "my costumer signed this

[3]The subscript here idicates the vector components.

and must pay that amount of money!!"). However, if the signature is obtained in a symmetric way there is no way in distinguishing the case in which someone sent me an authentic digital-signed document and the case in which I fake his signature by the means of the same key. This weakens the ability of 3rd party attestation.
To sum up, the first two points are achievable also in a classical regime with unconditional security, but the last one can provide only computational security.
How are they build in classical world? There is a sender A, some message u, a signing block S which changes $u \to x$, then x should be in principle received by a receiver B (imagine that this receives $\tilde{x}$) that performs a verification V. The output of the verification is an estimate of the original message $\hat{u}$ and a binary variable $\hat{b}$ stating the (expected) authenticity of the message (i.e., if B can consider it authentic or not). As always, we take into account possible imperfections of this pipeline and consider the worst case scenario where we have an attacker trying to violate the protocol. The attacker stands in the middle: he can have its forged messaged u' that he wants to substitute to the original one or simply change the upcoming one. Either way, we can model it as a new encoded message x' sent towards B. As receivers, we don't know whether we are getting an input from the legitimate user ($\tilde{x} = x$ or from the attacker ($\tilde{x} = x'$). Moreover, assume that an external provider sends to A and B the private and public keys, k and k' respectively. This allows the receiver to retrieve the original message from the input and and check its authenticity.
DISEGNO
The typical way to do this is the following: A sends x as actually a pair $x = (u, t)$ containing the message and a signature t. The signature t is obtained via some functions $t = T_k(u)$. Of course, B will receive the corresponding $\tilde{x} = (\tilde{u}, \tilde{t})$. B can perform verification by retrieving $\hat{u} = T'_{k'}(\tilde{t})$, where $T'_{k'} = T_k^{-1}$. Then, $\hat{b} = 0$ (authentic) if $\hat{u} = \tilde{u}$ and 1 otherwise. Of course, this is correct since k and k' are related (as they correctly relate T and T'). In particular, k is private and k' is public: therefore, one can simply generate at first k, calculate k' as a function $k' = f(k)$ and share it as public key. Notice that this implies that also the attacker knows the public key. In asymmetric encoding, every receiver could be an attacker (they all share the public key), so we have to think it this way.

2.1 Attacks

What are in concrete the most suitable attacks to this protocol?

1. **Forging**: the attacker is creating his own u' and tries to build a signed message $u' \mapsto x'$, such that $V_{k'}(x') = (u', \hat{b} = 0)$ (the verification test is passed and B accepts as authentic the attacker's message).

2. **Modification**: starting from A's message x, choose a modified version x' such that $V_{k'}(x') = (u' \neq u, \hat{b} = 0)$ (which is accepted). Here the attacker can change everything: the message, the signature.. it is just needed to create

something different (of course, hoping for verification to still be valid). Please notice that this might be not directly beneficial to the attacker in a rational way, but is something that can practically occur because of experimental imperfections (but we consider the worst case).

3. **Repudiation** A sends a message and then denies that she sent it.

We don't want them all to happen. Security countermeasures can be based on:

1. It must be hard (not impossible, think in terms of computational security) to derive k from k'. If $k' = f(k)$ from a function f, we must not be able to invert it: f is pre-image resistant (one-way: from k to k' but not vice versa).

2. It must the hard to derive x from (k', u), meaning that the verification function $V_{k'}$ $(T'_{k'})$ should also be one-way.

3. It must be hard to derive k from u, x, k'. Hence, A's signature $S(u, k)$ must be one way.

Countermeasures number 1 and 3 are the ones to be more careful about, since if they fail the attacker can obtain the key used to sign several systems! The security parameters in the classical scheme are the **length of the signature** $\ell_t = H(x|u)$ and the **key entropy** $\ell_k = H(k)$.
It is important to keep in mind the importance of one-way functions and the two different security parameters.

2.2 Quantum digital signatures

We must create some equivalent of one-way functions: **quantum unconditionally-secure one-way functions**!

$$k \mapsto |f_k\rangle \text{ where } k \in \mathbb{B}^L \text{ and } |f_k\rangle \in \bigotimes_{i=1}^{n} \mathcal{H}_i$$

The map must be such that $\forall\, \mathbf{k_1}, \mathbf{k_2} : \mathbf{k_1} \neq \mathbf{k_2}$**, it holds** $|\langle \mathbf{f_{k_1}} | \mathbf{f_{k_2}} \rangle|^{\mathbf{2}} \leq \delta$ (measure of distinguishability). If we require perfect orthogonality then $L \sim n$ (perfect correspondence between key bits and single particle states). It is important for δ to be < 1, since we will see how the security of this method depends on a power law of δ. If δ is large we will have to use more repetitions, i.e. account for more resources to be spent to increase security.
One-way means that it must be hard from $|f_k\rangle$ to retrieve the binary sequence k. From $|f_k\rangle$ one can obtain $\leq n$ bits of information, given the dimension of the state. Since $|f_k\rangle$ takes the role of the public key, it must be shared among all the subjects the sender wants to communicate with. Assume he/she shares many identical copies

of $|f_k\rangle$, a number T. From T of them the amount of bits of information that these people altogether can obtain about the private key is $\leq Tn$. So, we said that our function is one-way if $L \gg n$. We say that it is "public one-way" if $L \gg Tn$ (i.e, if a coalition of all the people we share $|f_k\rangle$ with cannot learn a meaningful amount of bits of the private key). Note that the last statement means that this protocol does **not allow infinitely possible receivers**!
To recap, the parameters of the protocol are the length of the private key (L), the dimension of the public key space (n) and the number of people we share the public key with (T).
Let us define the signature scheme for $u \in \mathbb{B}^1 = \{0, 1\}$ (**single bit message**). We have to define how we generate the private/public keys, how we sign and how we verify the bits.

1. **Key generation.** Firstly, choose a security parameter M (the larger it is the more secure is the mechanism, but also the more resource requiring). A generates at random M pairs $(k_{0,m},\ k_{1,m})$, where $k_{u,m} \in \mathbb{B}^L$. From each of this it computes with the one-way function the public (quantum) key $\left|f_{k_{u,m}}\right\rangle$. Note that the mapping $k \mapsto |f_k\rangle$ is publicly known. A distributes a copy of the $\left|f_{k_{u,m}}\right\rangle$ to each receiver.

2. **Signing.** Like in the classical case, A creates $x = (u, t)$, where the signature t is the sequence of key values that corresponds to the bit u: $t = (k_{u,1}, \ldots, k_{u,m})$. A sends x to the receivers.

3. **Verifying.** B receives $\tilde{x} = (\tilde{u}, \tilde{t})$, where $\tilde{t} = (\tilde{k}_1, \ldots, \tilde{k}_m)$. Since B knows the one-way mapping, he can construct the quantum state that corresponds to the signature he received (i.e, $\left|f_{\tilde{k}_m}\right\rangle$) and check if $\left|f_{\tilde{k}_m}\right\rangle = \left|f_{k_{\tilde{u},m}}\right\rangle$, i.e. the valid signature A provided him corresponding to index $\tilde{u}$. These two states must be exactly the same $\forall m = 1, \ldots, M$. In fact, if $x = \tilde{x}$ (everything is correct), then $u = \tilde{u}$, $\tilde{k}_m = k_{\tilde{u},m}$ and therefore also the ket states are. How do we check whether these are equal? Multiple possibilities: swap test or project one on the other for example, but even a destructive test! So by comparing the quantum states one can calculate the set of b_m and if $b_m = 0\,\forall m$ this is the good case (accepted verification).

What if the latter is not true? By the rules of quantum mechanics the probability of having the same quantum states is:

$$P[b_m = 0] = |\left\langle f_{\tilde{k}_m} \middle| f_{k_{\tilde{u},m}} \right\rangle|^2$$

And therefore for the whole set:

$$P[\vec{b}=\vec{0}] = \Pi_{m=1}^{M}\left(P[b_m=0]\right) =$$

$$\Pi_{m=1}^{M}\left(P[b_m=0|\tilde{k}_m\neq k_{\tilde{u},m}]P[\tilde{k}_m\neq k_{\tilde{u},m}] + \underbrace{P[b_m=0|\tilde{k}_m=k_{\tilde{u},m}]}_{=1}P[\tilde{k}_m=k_{\tilde{u},m}]\right)$$

Where by construction if $\tilde{k}_m = k_{\tilde{u},m}$ then $b_m = 0$ (same quantum states). Please notice that $\tilde{k}_m$ is indeed provided by the attacker, since it is what B receives. As a result, we cannot control its distribution, but only the one of $k_{\tilde{u},m}$. In the most general sense, $\tilde{k}_m, k_{\tilde{u},m}$ are L-long bit strings, hence the probability of them being equal is $1/2^L$. Moreover, the term $P[b_m=0|\tilde{k}_m\neq k_{\tilde{u},m}]$ is given by the quantum overlap $|\left\langle f_{\tilde{k}_m}\middle| f_{k_{\tilde{u},m}}\right\rangle|^2$ with $\tilde{k}_m \neq k_{\tilde{u},m}$, which we required to be $\leq \delta$. Putting everything together it results:

$$P[\vec{b}=\vec{0}] \leq \Pi_{m=1}^{M}\left(\delta\cdot\left(1-\frac{1}{2^L}\right)+1\cdot\frac{1}{2^L}\right) < \left(\delta+\frac{1}{2^L}\right)^M \equiv \varepsilon$$

Therefore, if $\delta < 1$ this term goes exponentially to zero with M. δ, L and M are the parameters which determine the security of the protocol.
Notice how this method is indeed highly inefficient: after all this procedure, we delivered an authentic message of a single bit and we are throwing away a large set of auxiliary information.

2.3 Problems

It is essential that the quantum sates distributed at the beginning are distributed securely, meaning that either Bob and any other trusted party must hold the same public states. So, all T copies of $|f_k\rangle$ must be identical (within the quantum permissions for this). This can be checked by performing non-destructive swap tests between the sets of public key qubits sent to different parties (trusted distribution center).
Another issue is the fact that the qubits need to be stored until the rest of the protocol is carried out: this is in principle problematic since we do not yet have quantum memories. A solution which was proposed under the name "QDS without quantum memories" modifies the protocol by considering public states σ_0, σ_1 defined as

$$\sigma_u = \bigotimes_{\ell=1}^{L}\rho_{u,\ell} \text{ where } \rho_{u,\ell} = |b_{u,\ell}\alpha\rangle\langle b_{u,\ell}\alpha|$$

Where $b_{u,\ell}$ i.i.d. drawn from $\mathcal{U}(\{-1,1\})$ is a random bit and $\alpha \in \mathbb{R}^+$ (therefore, the latter are signed coherent states). Then, similarly to the previous case, the roles of $k_{u,m}$ are taken by $b_{u,\ell}$ and therefore the private keys are defined by the bit strings

$\vec{b}_0 = (b_{0,1}, \ldots, b_{0,L})$ and $\vec{b}_1 = (b_{1,1}, \ldots, b_{1,L})$. Now, consider two legitimate receivers Bob and Charlie that get their public states from Alice. Since these are coherent states, they can make them interfere in a Mach-Zender fashion by means of BSs as displayed in figure 12.1.

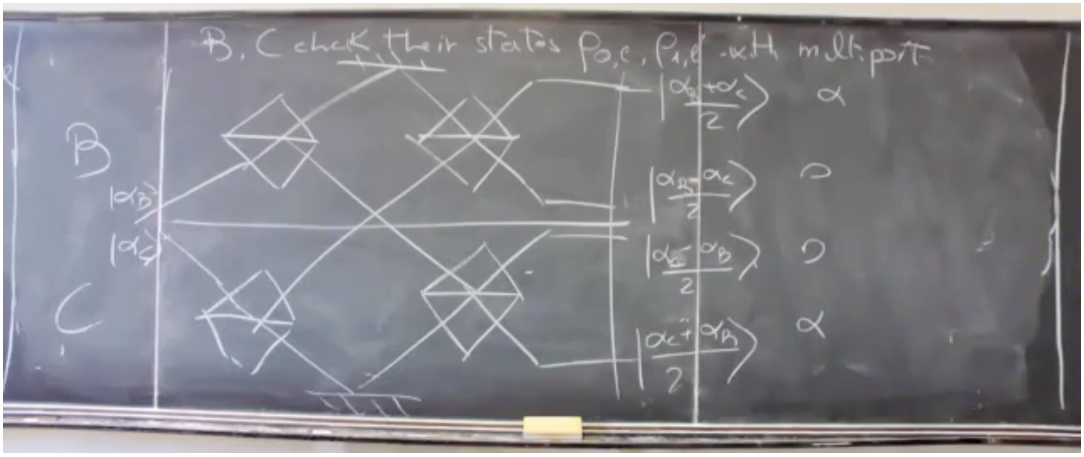


Figure 12.1: QDS without quantum memories

The final exits are by interference composed as follows: the first and last with the state $|(\alpha_B + \alpha_C)/2\rangle$, the others with $|(\alpha_B - \alpha_C)/2\rangle$. As a result, if everything works properly and $\alpha_B = \alpha_C$ then lights exits only from one port for each party (signal port), otherwise light is seen on a control port. This allows also to detect A's repudiation. In this case, with reference to the notation used above, it holds $x = \left(u, \vec{b}_u\right)$.
Another idea is to exploit a BB84-like approach, with $\rho_{u,\ell} = |b_{u,\ell}\rangle\langle b_{u,\ell}|$ where $|b_{u,\ell}\rangle \in \{|0\rangle, |1\rangle, |+\rangle, |-\rangle\}$ and Bob and Charlie perform projective measurements in the BB84 bases.

Bibliography

[Herrero-Collantes and Garcia-Escartin(2017)] M. Herrero-Collantes and J. C. Garcia-Escartin, Rev. Mod. Phys. **89**, 015004 (2017).

[Konig *et al.*(2009)Konig, Renner, and Schaffner] R. Konig, R. Renner, and C. Schaffner, IEEE Transactions on Information Theory **55**, 4337 (2009).

[Stanco *et al.*(2020)Stanco, Marangon, Vallone, Burri, Charbon, and Villoresi] A. Stanco, D. G. Marangon, G. Vallone, S. Burri, E. Charbon, and P. Villoresi, Phys. Rev. Research **2**, 023287 (2020).

[Fiorentino *et al.*(2007)Fiorentino, Santori, Spillane, Beausoleil, and Munro] M. Fiorentino, C. Santori, S. M. Spillane, R. G. Beausoleil, and W. J. Munro, Phys. Rev. A **75**, 032334 (2007).

[Tomamichel and Renner(2011)] M. Tomamichel and R. Renner, Physical Review Letters **106** (2011), 10.1103/physrevlett.106.110506.

[Cao *et al.*(2015)Cao, Zhou, and Ma] Z. Cao, H. Zhou, and X. Ma, New Journal of Physics **17**, 125011 (2015).

[Lunghi *et al.*(2015)Lunghi, Brask, Lim, Lavigne, Bowles, Martin, Zbinden, and Brunner] T. Lunghi, J. B. Brask, C. C. W. Lim, Q. Lavigne, J. Bowles, A. Martin, H. Zbinden, and N. Brunner, Phys. Rev. Lett. **114**, 150501 (2015).

[Pironio *et al.*(2010)Pironio, Acín, Massar, de la Giroday, Matsukevich, and et al.] S. Pironio, A. Acín, S. Massar, A. B. de la Giroday, T. A. Matsukevich, and et al., Nature **464**, 1021–1024 (2010).

[Shaltiel(2011)] R. Shaltiel, in *International colloquium on automata, languages, and programming* (Springer, 2011) pp. 21–41.

[Trevisan(2001)] L. Trevisan, J. ACM **48**, 860–879 (2001).

[Raz *et al.*(2002)Raz, Reingold, and Vadhan] R. Raz, O. Reingold, and S. Vadhan, Journal of Computer and System Sciences **65**, 97 (2002).

[De *et al.*(2012)De, Portmann, Vidick, and Renner] A. De, C. Portmann, T. Vidick, and R. Renner, SIAM Journal on Computing **41**, 915–940 (2012).

[Tomamichel *et al.*(2011a)Tomamichel, Schaffner, Smith, and Renner] M. Tomamichel, C. Schaffner, A. Smith, and R. Renner, IEEE Transactions on Information Theory **57**, 5524 (2011a).

[Tomamichel *et al.*(2011b)Tomamichel, Schaffner, Smith, and Renner] M. Tomamichel, C. Schaffner, A. Smith, and R. Renner, IEEE Transactions on Information Theory **57**, 5524 (2011b).

[Wang *et al.*(2018)Wang, Yu, and Hu] X.-B. Wang, Z.-W. Yu, and X.-L. Hu, Phys. Rev. A **98**, 062323 (2018).

[Hu *et al.*(2019)Hu, Jiang, Yu, and Wang] X.-L. Hu, C. Jiang, Z.-W. Yu, and X.-B. Wang, Phys. Rev. A **100**, 062337 (2019).

[Duan *et al.*(2001)Duan, Lukin, Cirac, and Zoller] L. Duan, M. Lukin, J. Cirac, and P. Zoller, Nature **414**, 413 (2001).

[Lo and Chau(1997)] H.-K. Lo and H. F. Chau, Phys. Rev. Lett. **78**, 3410 (1997).

[Mayers(1997)] D. Mayers, Phys. Rev. Lett. **78**, 3414 (1997).

[Konig *et al.*(2012)Konig, Wehner, and Wullschleger] R. Konig, S. Wehner, and J. Wullschleger, IEEE Transactions on Information Theory **58**, 1962 (2012).

[Kaniewski *et al.*(2013)Kaniewski, Tomamichel, Hanggi, and Wehner] J. Kaniewski, M. Tomamichel, E. Hanggi, and S. Wehner, IEEE Transactions on Information Theory **59**, 4687–4699 (2013).

[Gama *et al.*(2020)Gama, Mateus, and Souto] M. Gama, P. Mateus, and A. Souto, Entropy **22**, 272 (2020).