Introduction to Quantum Information

André Chailloux

Contents

Foreword

These lectures notes are intended for Masters students of Sorbonne Université attending the course Introduction to Quantum Information. This course follows to the Quantum Circuits and Logic Gates. If you have difficulties understanding some material in these lecture notes, a good thing to do is to read some other lecture notes where things will be explained in a different manner and maybe you will get a key information that wasn't here. I strongly recommend Ronald de Wolf's lecture notes[1] which cover most of the topics we will present here and are very well written. You can also check the book *Quantum Information and Quantum Computation* by Nielsen and Chuang which is still the reference textbook for quantum computing. These lecture notes are written on the fly for the course of winter/spring 2022. There will probably be some typos and mistakes (hopefully not too many) in the first iterations of these lecture notes. Remarks, comments on these lecture notes are very welcome, particularly if you find some typos or mistakes. You can contact me at `andre.chailloux@inria.fr`.

[1] `https://homepages.cwi.nl/~rdewolf/qcnotes.pdf`

Chapter 1

The quantum Fourier transform

The quantum Fourier will be one of our main tools when constructing quantum algorithms. It will be at the heart of Shor's factoring algorithm.

1.1 The classical Fourier transform

Widely used: data compression, signal processing, complexity theory. Here, we will consider only the discrete Fourier transform.

1.1.1 Definition

Fourier transform F_N: $N \times N$ unitary matrix, with elements of the same magnitude.

$$F_2 := H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}.$$

$N = 3$ (for example): impossible to achieve with real numbers. Use complex numbers. We will use roots of unity $\omega_N = e^{\frac{2i\pi}{N}}$. The discrete Fourier transform F_N is defined as

$F_N := \frac{1}{\sqrt{N}} \begin{pmatrix} \ddots & \vdots & \iddots \\ \dots & \omega_N^{jk} & \dots \\ \iddots & \vdots & \ddots \end{pmatrix}$ meaning that for any line $j \in \{0, \dots, N-1\}$ and any column $k \in \{0, \dots, N-1\}$, we have $(F_N)_{jk} := \omega_N^{jk}$.

Properties

- Each column $C_k = \frac{1}{\sqrt{N}} \begin{pmatrix} 1 \\ \omega_N^k \\ \vdots \\ \omega_N^{(N-1)k} \end{pmatrix}$ has norm 1 and any two columns are orthogonal. Indeed

$$\forall k, k' \in \{0, \dots, N-1\}, \ \frac{1}{N} \sum_{j=0}^{N-1} \omega_N^{jk} \omega_N^{-jk'} = \frac{1}{N} \sum_{j=0}^{N-1} \omega_N^{j(k-k')} = \delta_{kk'}.$$

1.1.2 Computing the classical Fourier transform

Computing the classical Fourier transform problem

Input: a column vector $v = \begin{pmatrix} v_0 \\ \vdots \\ v_{N-1} \end{pmatrix}$.

Goal: compute $\hat{v} = F_N \cdot v$.

Naive way: Perform the whole multiplication entry-wise

- $O(N)$ operations ($+$ and $\times$) per entry.
- $O(N^2)$ operations in total.

Fast Fourier transform

- $O(N \log(N))$ operations in total.
- recursive algorithm.

1.2 The quantum Fourier transform

1.2.1 Definition of the problem

Take $N = 2^n$. Since F_N is a $N \times N$ unitary matrix, we can interpret it as a quantum unitary operation acting on n qubits.

Computing the quantum Fourier transform problem

Input: a quantum state $|\psi\rangle$ of n qubits.
Goal: output $F_N(|\psi\rangle)$.

How efficiently can we implement this quantum Fourier transform?

- *QFT* can be implemented with a quantum circuit of size $O(n^2)$. The rest of the chapter will be devoted to the construction of this algorithm.
- Exponentially faster than classical FFT which runs in $O(N \log(N))$.

⚠ In the classical setting, we are given an explicit (written) description of a vector $v = \begin{pmatrix} v_0 \\ \vdots \\ v_{N-1} \end{pmatrix}$ as an input and ask to have a similar description of the output. In the quantum setting, we are given a quantum state $|\psi\rangle = \sum_{i=0}^{N-1} v_i \,|i\rangle$ and ask to output the state

$F_N(|\psi\rangle) = \sum_{i=0}^{N-1} \hat{v}_i |i\rangle$. Notice that we cannot fully recover the vector $\hat{v} = \begin{pmatrix} \hat{v}_0 \\ \vdots \\ \hat{v}_{N-1} \end{pmatrix}$ from $F_N(|\psi\rangle)$.

Even though the speedup is exponential, it doesn't allow to recover $\hat{v}$, which makes it incomparable with the FFT. The quantum algorithm in particular doesn't help in computing the classical Fourier transform. It will still have other important uses.

1.2.2 Efficient quantum circuit for QFT

Elementary gates

- Hadamard gate $H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$.
- Phase rotation $R_s = \begin{pmatrix} 1 & 0 \\ 0 & e^{\frac{2i\pi}{2^s}} \end{pmatrix}$.
- Controlled R_s gate written C-R_s and acting on 2 qubits.

  $$C\text{-}R_s(|0\rangle|x\rangle) := |0\rangle|x\rangle \quad ; \quad C\text{-}R_s(|1\rangle|x\rangle) := |1\rangle R_s(|x\rangle) \quad \text{which gives} \quad C\text{-}R_s(|b\rangle|x\rangle) = |b\rangle e^{\frac{2i\pi bx}{2^s}}|x\rangle .$$

For each k, we have $F_N(|k\rangle) = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \omega_N^{jk} |j\rangle$. For any integer j, we write its binary decomposition $j = j_1, \dots, j_n$ where j_1 is the bit of highest weight. This means we can write $j = \sum_{l=1}^{n} 2^{n-l} j_l$. We have

$$\begin{aligned} F_N(|k\rangle) &= \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} e^{\frac{2i\pi jk}{2^n}} |j\rangle \\ &= \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} e^{2i\pi(\sum_{l=1}^{n} j_l 2^{-l})k} |j_1, \dots, j_n\rangle \\ &= \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \Pi_{l=1}^{n} e^{2i\pi j_l 2^{-l} k} |j_1, \dots, j_n\rangle \\ &= \bigotimes_{l=1}^{n} \frac{1}{\sqrt{2}} \left(|0\rangle + e^{2i\pi k 2^{-l}} |1\rangle \right) \end{aligned}$$

To prove the last equality, recall that the tensor product satisfies

$$(\alpha|0\rangle + \beta|1\rangle) \otimes (\alpha'|0\rangle + \beta'|1\rangle) = \alpha\alpha'|00\rangle + \alpha\beta'|01\rangle + \beta\alpha'|10\rangle + \beta\beta'|11\rangle .$$

For any integer k, with binary decomposition $k = k_1, \dots, k_n$, we define $0.k := \frac{k}{2^n} = \sum_{l=1}^{n} k_l 2^{-l}$. For example, $0.010 = \frac{1}{4}$ and $0.101 = \frac{5}{8}$. Notice that

$$\begin{aligned} e^{2i\pi k 2^{-l}} &= e^{2i\pi(\sum_{m=1}^{n} 2^{n-m} k_m) 2^{-l}} \\ &= e^{2i\pi \left(\sum_{m=n-l+1}^{n} k_m 2^{m-(n-l+1)}\right) 2^{-l}} \\ &= e^{2i\pi \left(\sum_{m'=1}^{l} k_{n-m'+1} 2^{-m'}\right)} \\ &= 0.k_n \dots k_{n-l+1}. \end{aligned}$$

The second equality uses the fact $e^{2i\pi C} = 1$ for any $C \in \mathbb{N}$ which implies that only the $l-1$ bits of k of least weight matter in the term $e^{2\mathrm{i}\pi k 2^{-l}}$. We can therefore rewrite

$$F_N(|k\rangle) = \bigotimes_{l=1}^{n} \frac{1}{\sqrt{2}} \left(|0\rangle + e^{2\mathrm{i}\pi k 2^{-l}} |1\rangle \right) = \frac{1}{\sqrt{2}}(|0\rangle + e^{2\mathrm{i}\pi 0.k_n} |1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle + e^{2\mathrm{i}\pi 0.k_{n-1}k_n} |1\rangle) \otimes \cdots \otimes \frac{1}{\sqrt{2}}(|0\rangle + e^{2\mathrm{i}\pi 0.k_1 \ldots k_n} |1\rangle$$

The $n = 3$ ($N = 8$) case

From the above, we have

$$F_8(|k_1 k_2 k_3\rangle) = \frac{1}{\sqrt{2}}(|0\rangle + e^{2\mathrm{i}\pi 0.k_3} |1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle + e^{2\mathrm{i}\pi 0.k_2 k_3} |1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle + e^{2\mathrm{i}\pi 0.k_1 k_2 k_3} |1\rangle).$$

Because of this product structure, we can easily construct each qubit separately. Notice that with the $0.k$ notation, we can write $C\text{-}R_s(|b\rangle |x\rangle) = |b\rangle e^{\frac{2\mathrm{i}\pi bx}{2^s}} |x\rangle = |b\rangle e^{2\mathrm{i}x\pi \cdot 0.0 \ldots 0b} |x\rangle$. where $0.\underbrace{0 \ldots 0b}_{s \text{ bits}}$.

1st qubit:

$$|k_3\rangle \xrightarrow{H} \frac{1}{\sqrt{2}}(|0\rangle + (-1)^{k_3} |1\rangle) = \frac{1}{\sqrt{2}}(|0\rangle + e^{2\mathrm{i}\pi 0.k_3} |1\rangle).$$

2nd qubit:

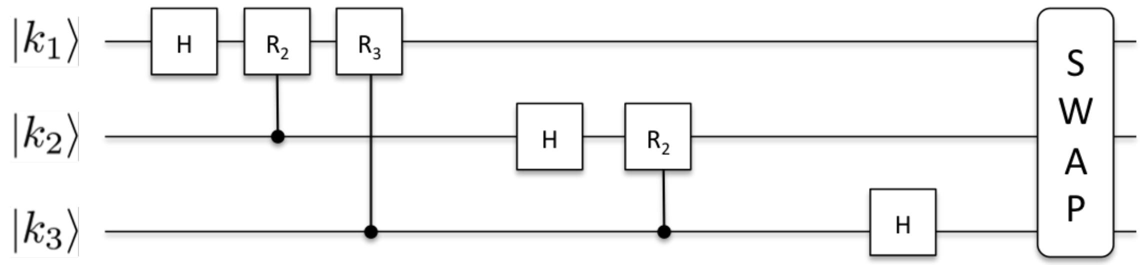
$$|k_2\rangle \xrightarrow{\mathrm{H}} \frac{1}{\sqrt{2}}(|0\rangle + (-1)^{k_2} |1\rangle) = \frac{1}{\sqrt{2}}(|0\rangle + e^{2\mathrm{i}\pi 0.k_2} |1\rangle)$$
$$|k_3\rangle \frac{1}{\sqrt{2}}(|0\rangle + e^{2\mathrm{i}\pi 0.k_2} |1\rangle) \xrightarrow{C\text{-}R_2} |k_3\rangle \frac{1}{\sqrt{2}}(|0\rangle + e^{2\mathrm{i}\pi 0.k_2 k_3} |1\rangle)$$

3rd qubit:

$$|k_1\rangle \xrightarrow{\mathrm{H}} \frac{1}{\sqrt{2}}(|0\rangle + (-1)^{k_1} |1\rangle) = \frac{1}{\sqrt{2}}(|0\rangle + e^{2\mathrm{i}\pi 0.k_1} |1\rangle)$$
$$|k_2\rangle \frac{1}{\sqrt{2}}(|0\rangle + e^{2\mathrm{i}\pi 0.k_1} |1\rangle) \xrightarrow{C\text{-}R_2} |k_2\rangle \frac{1}{\sqrt{2}}(|0\rangle + e^{2\mathrm{i}\pi 0.k_1 k_2} |1\rangle)$$
$$|k_3\rangle \frac{1}{\sqrt{2}}(|0\rangle + e^{2\mathrm{i}\pi 0.k_1 k_2} |1\rangle) \xrightarrow{C\text{-}R_3} |k_3\rangle \frac{1}{\sqrt{2}}(|0\rangle + e^{2\mathrm{i}\pi 0.k_1 k_2 k_3} |1\rangle)$$

The 3rd qubit is stored in the quantum register where $|k_1\rangle$ is and uses $|k_2\rangle$ and $|k_3\rangle$ as control qubits. The 2nd qubit is be stored in the quantum register where $|k_2\rangle$ is and uses $|k_3\rangle$ as a control qubit. The 1st qubit is stored where $|k_3\rangle$ is. In order to do this, we must first construct the 3rd qubit, then the 2nd and finally the 1st qubit. In order to have the good ordering of qubits, we end up inverting the order of all the qubits.

The circuit of F_8 is the following



The number of gates used here is $3 + 2 + 1$ gates + the gates in the SWAP.

General case

The construction for $n = 3$ can be extended to any n following the same pattern. The first qubit will consist only of an H gate while the last qubit will require applying H, $C\text{-}R_2, \ldots, C\text{-}R_n$. Similarly as in the $n = 3$ case, we finish by inverting the order of all the qubits. The total number of gates used is therefore $n+n-1+\cdots+1+\text{SWAP} = O(n^2)$gates.

Improvements: we can reduce the number of gates if we allow for some small errors:

- As s grows, $C\text{-}R_s$ fastly converges to the identity gate.
- We remove all those gates for $s \geq \Omega(\log(n))$. By a careful error analysis, we can show that the result will be $O(\frac{1}{poly(n)})$ close to the desired state.
- The total amount of needed gates becomes $O(n \log(n))$ (SWAP is $O(n)$).

1.3 Phase estimation

Our first protocol is a direct application of the quantum Fourier transform.

Phase estimation

Input: a quantum unitary U acting on n qubits. An eigenvector $|\psi\rangle$ of U with eigenvalue λ given as a quantum state.
Goal: output λ.

Recall that an eigenvector $|\psi\rangle$ of U with eigenvalue λ means that $U(|\psi\rangle) = \lambda |\psi\rangle$. Because U is a unitary, $|\lambda| = 1$ so we can write $\lambda = e^{2i\pi\phi}$ for some real number $\phi \in [0, 1)$ ($[0, 1[$ in French notation). We assume first that ϕ can be fully described with l bits of precision, *i.e.* there exists a natural number $C \in \mathbb{N}$ such that $\phi = \frac{C}{2^l}$.

We consider a quantum unitary Q satisfying

$$Q(|k\rangle |\psi\rangle) = |k\rangle U^k(|\psi\rangle).$$

for any $k \in \{0, \ldots, 2^l - 1\}$ and any state $|\psi\rangle$. We perform the following algorithm:

1. Start from $|0\rangle\,|\psi\rangle$ and apply F_{2^l} on the first register. The resulting state is
$$\frac{1}{\sqrt{2^l}}\sum_{k=0}^{2^l-1}|k\rangle\,|\psi\rangle\,.$$
2. Apply Q on both registers.
$$\begin{aligned}\frac{1}{\sqrt{2^l}}\sum_{k=0}^{2^l-1}U^k\,|k\rangle\,|\psi\rangle &= \frac{1}{\sqrt{2^l}}\sum_{k=0}^{2^l-1}\lambda^k\,|k\rangle\,|\psi\rangle\\ &= \frac{1}{\sqrt{2^l}}\sum_{k=0}^{2^l-1}e^{\frac{2\mathrm{i}\pi kC}{2^l}}\,|k\rangle\,|\psi\rangle\,.\end{aligned}$$
3. Apply the inverse Fourier transform $F_{2^l}^{-1}$ on the first register. The resulting state is $|C\rangle\,|\psi\rangle$. It is then easy to recover ϕ from C.

General case If ϕ cannot be written with l bits of precision, we consider the closest approximation of ϕ of the form $\frac{C}{2^l}$. An error analysis (not detailed here) shows that the above procedure will find this C with probability at least $\frac{4}{\pi^2}$. By performing several iterations of this procedure, we can find the correct C, *i.e.* a good approximation of ϕ with a probability that exponentially converges to 1 in the number of iterations.

Efficiency of the algorithm If U can be computed efficiently and if, for any $k \in \{0,\ldots,2^l-1\}$, U^k can be computed efficiently then Q can be computed efficiently and the whole algorithm is efficient. This means that if we want our algorithm to run in time *poly*(n) (assuming U can be computed in time poly(n)), we have to take $l = O(\log(n))$.leq

1.4 Application: Fourier transform F_N for any $N \in \mathbb{N}$

In Chapter 1, we showed how to perform the Fourier transform F_N when $N = 2^n$ for some $n \in \mathbb{N}$. Here, we show how to perform the Fourier transform for any N. F_N will act on a quantum register that can take N values from 0 to $N-1$ and
$$\forall k \in \{0,\ldots,N-1\},\ F_N(|k\rangle) = \frac{1}{\sqrt{N}}\sum_{j=0}^{N-1}\omega^{jk}\,|j\rangle\,.$$
where $\omega := e^{\frac{2\mathrm{i}\pi}{N}}$. Let U_1 and U_2 two unitaries that do the following, $\forall k \in \{0,\ldots,N-1\}$.
$$U_1(|k\rangle\,|0\rangle) = |k\rangle\,F_N(|k\rangle) \quad ; \quad U_2(F_N\,|k\rangle\,|0\rangle) = F_N(|k\rangle)\,|k\rangle\,.$$
From those two unitaries, we can perform F_N as follows
$$|k\rangle\,|0\rangle \xrightarrow{U_1} |k\rangle\,F_N(|k\rangle) \xrightarrow{SWAP} F_N(|k\rangle)\,|k\rangle \xrightarrow{U_2} F_N(|k\rangle)\,|0\rangle\,.$$
what is left to show is how to perform U_1 and U_2. U_1 can be performed quite easily.

Let S_N a quantum unitary such that $S_N(|0\rangle) = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} |j\rangle$. Since N is not a power of 2, S_N cannot be expressed as Hadamards but we can still easily construct such a unitary. Let also O_{mult} satisfying $O_{mult}(|k\rangle |j\rangle |0\rangle) = |k\rangle |j\rangle |kj \mod N\rangle$.

Let's now construct U_1.

1. Start from $|k\rangle |0\rangle |0\rangle$ and apply S_N on the second register. The resulting state is
$$|k\rangle \frac{1}{\sqrt{N}} \sum_j |j\rangle |0\rangle .$$
2. Apply O_{mult} on the three registers.
$$|k\rangle \frac{1}{\sqrt{N}} \sum_j |j\rangle |kj \mod N\rangle .$$
3. Apply the unitary $|x\rangle \to \omega^x |x\rangle$ on the third register.
$$|k\rangle \frac{1}{\sqrt{N}} \sum_j \omega^{kj} |j\rangle |kj \mod N\rangle .$$
4. Apply O_{mult}^{-1} on the three registers. We obtain
$$|k\rangle \frac{1}{\sqrt{N}} \sum_j \omega^{kj} |j\rangle |0\rangle = |k\rangle F_N(|k\rangle) |0\rangle .$$

Swapping registers is then easy. What is left to do is to perform unitary U_2. We consider the unitary O_{add} such that $O_{add}(|k\rangle) = |k+1 \mod N\rangle$. The idea is that $F_N(|k\rangle)$ is a eigenvector of O_{add} with eigenvalue λ_k and that it will be easy to recover k from λ_k. Doing this in a coherent way will give U_2.

First see that
$$O_{add} F_N(|k\rangle) = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \omega^{jk} |j+1 \mod N\rangle = \omega^{-k} \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \omega^{jk} |j\rangle$$
which means that $F_N(|k\rangle)$ is an eigenvector of O_{add} with eigenvalue $\omega^{-k} = \omega^{N-1-k}$. If we apply the phase estimation from the previous section, with $l = \lceil \log(N) \rceil$, we obtain
$$F_N(|k\rangle) |0\rangle \xrightarrow{\text{Phase Estimation with } O_{add}} F_N(|k\rangle) |N-1-k\rangle .$$
Which by applying $|N-1-x\rangle \to |x\rangle$ gives the correct result. Actually, the phase estimation will only give $N-k$ with some (high) probability so this whole process will construct an approximation of U_2.

Putting everything together, we managed to construct $F_N(|k\rangle)$. This unitary has many applications, for example for the Discrete Log quantum algorithm.

Chapter 2

Shor's quantum factoring algorithm

Shor's idea:

- There exists an efficient quantum algorithm for finding the period of a function.
- Factoring can be reduced to period finding *i.e.* an efficient algorithm for period finding $\Rightarrow$ an efficient algorithm for factoring.

Period finding problem

Input: a function $f : \mathbb{N} \to \{0, \dots, N-1\}$ such that $\exists r \in \{0, \dots, N-1\}$ (unknown) such that
$f(a) = f(b) \Leftrightarrow a = b \mod r$.
Goal: output r.

2.1 From factoring to period finding

2.1.1 Classical algorithm for factoring a number N using period finding

Equivalent to finding a non trivial factor of N.

1. Pick a random $x \in \{2, \dots, N-1\}$.
2. Calculate $x \wedge N$ (efficient, use Euclid's algorithm).
   - if $x \wedge N = c \neq 1 \to c$ divides N.
   - if $x \wedge N = 1 \to$ continue.
3. Consider the smallest $r \in \{0, \dots, N-1\}$ such that $x^r = 1 \mod N$. Since $x \wedge N = 1$, such an r exists.
4. r is the period of the function $f(k) = x^k \mod N$. Use the period finding algorithm to find r. If r is odd, go back to step 1.

5. Calculate $(x^{r/2}+1) \wedge N$ and $(x^{r/2}-1) \wedge N$. If one of those values is different than 1 or N then this value is a non trivial factor of n. If both of those values are equal to 1 or N, start again from step 1.

2.1.2 Proof that the algorithm works

The main part of the proof will be the following lemma from number theory. The proof will be omitted.

Lemma 1. *For any odd N, for a randomly chosen x such that $x \wedge N = 1$ and r begin the smallest element in $\{0, \dots, n-1\}$ satisfying $x^r = 1 \mod N$, the event*

$$E: \quad r \textit{ is even} \quad \wedge \quad (x^{r/2}+1) \neq 0 \mod N \quad \wedge \quad (x^{r/2}+1) \neq 0 \mod N$$

occurs with probability $\geq \frac{1}{2}$.

If r is even, we have

$$\begin{aligned} x^r = 1 \mod N &\Leftrightarrow (x^{r/2}+1)(x^{r/2}-1) = 0 \mod N \\ &\Leftrightarrow \exists k \in \mathbb{N}^*,\ (x^{r/2}+1)(x^{r/2}-1) = kN. \end{aligned}$$

Notice first that $(x^{r/2}+1) > 0$ and we also have $(x^{r/2}-1) > 0$ because $x \geq 2$ and $r \geq 2$. If E holds, both $(x^{r/2}+1)$ and $(x^{r/2}-1)$ are not multiples of N. Therefore, they will both have a non trivial factor of N and we actually have $(x^{r/2}-1) \wedge N \neq 1$ and $(x^{r/2}+1) \wedge N \neq 1$. Conclusion: if E holds then step 5 outputs a non trivial factor of N. Since $\Pr[E] \geq \frac{1}{2}$, we require $O(1)$ calls to the period finding algorithm for the algorithm to succeed with a high (constant) probability.

2.2 Shor's period finding algorithm

Our goal here is to present Shor's quantum algorithm for period finding. Let $n := \lceil \log(N) \rceil$, $q := \lceil \log(N^2) \rceil$ and $Q := 2^q \in [N^2, 2N^2]$. We have a quantum access to $f : \mathbb{N} \to \{0, \dots, N-1\}$. We restrict the input space to q input bits and consider the quantum unitary

$$O_f : |x\rangle_q |0\rangle_n \to |x\rangle_q |f(x)\rangle_n .$$

The subscripts represent the number of qubits in each register. This means for example that register $|x\rangle_q$ contains q qubits and register $|0\rangle_n$ contains n qubits.

2.2.1 Algorithm for period finding

1. Initialize the protocol at the state

$$|0\rangle_q |0\rangle_n .$$

2. Apply QFT_Q on the first register. We get

$$\frac{1}{\sqrt{Q}} \sum_{a=0}^{Q-1} |a\rangle_q |0\rangle_n .$$

3. Apply O_f on the whole state to obtain

$$\frac{1}{\sqrt{Q}}\sum_{a=0}^{Q-1} |a\rangle_q |f(a)\rangle_n .$$

4. Measure the second register: it gives some value $f(s)$ for some $s < r$. Let $m := \#\{a \in \{0,\dots,Q-1\} : f(a) = f(s)\}$. We have

$$\begin{aligned}\{a \in \{0,\dots,Q-1\} : f(a) = f(s)\} &= \{s, s+r, \dots, s+(m-1)r\}\\ &= \{jr+s\}_{0\leq j<m}\end{aligned}$$

When measuring $f(s)$ in the second register, the first register collapses to

$$\frac{1}{\sqrt{m}}\sum_{j=0}^{m-1} |jr+s\rangle .$$

5. Apply QFT_Q on this first register.

$$\begin{aligned}&\frac{1}{\sqrt{m}}\sum_{j=0}^{m-1}\frac{1}{\sqrt{Q}}\sum_{b=0}^{Q-1} e^{\frac{2i\pi(jr+s)b}{Q}} |b\rangle\\ &= \frac{1}{\sqrt{mQ}}\sum_{b=0}^{Q-1} e^{\frac{2i\pi sb}{Q}}\left(\sum_{j=0}^{m-1} e^{\frac{2i\pi jrb}{Q}}\right)|b\rangle\end{aligned}$$

6. Measure the first register. What is the probability of outputting each specific b?

Special case developed here : r divides Q.

In this case, we have $m = \frac{Q}{r}$. We have

$$b \text{ is a multiple of } \frac{Q}{r} \Leftrightarrow e^{\frac{2i\pi rb}{Q}} = 1.$$

Any such b will therefore have squared amplitude

$$\begin{aligned}&\Big|\frac{1}{\sqrt{mQ}}e^{\frac{2i\pi sb}{Q}}\left(\sum_{j=0}^{m-1} e^{\frac{2i\pi jrb}{Q}}\right)\Big|^2\\ &= \Big|\frac{1}{\sqrt{mQ}}e^{\frac{2i\pi sb}{Q}}\left(\sum_{j=0}^{m-1} 1\right)\Big|^2\\ &= \frac{m}{Q} = \frac{1}{r}\end{aligned}$$

Each $b \in \{0,\dots,Q-1\}$ which is a multiple of $\frac{Q}{r}$ will be measured with probability exactly $\frac{1}{r}$. Notice also that there are exactly r such multiples, which are the elements of $\{0, \frac{Q}{r}, \dots, \frac{(r-1)Q}{r}\}$. Therefore, the measurement will always output a multiple of $\frac{Q}{r}$.

$$\text{Output } b : \text{a uniformly random multiple of } \frac{Q}{r}.$$

This means there exists a random (unknown) $c \in \{0, \dots, r-1\}$ such that $b = \frac{cQ}{r}$. or equivalently $\frac{b}{Q} = \frac{c}{r}$.

7. Find r from the above equality. How?
    - b, Q are known, c, r are unknown. We can rewrite $\frac{b}{Q} = \frac{b'}{Q'}$ with $b' \wedge Q' = 1$.
    - c is a random number in $\{0, \dots, r-1\}$. This implies that $c \wedge r = 1$ with probability greater than $\Omega(\frac{1}{\log(\log(r))})$. When this happens, we necessarily have $c = b'$ and $r = Q'$.
    - Check that Q' is a period of f. If yes: done. If no, go back to step 1.

General case (Sketch) : r does not divide Q.

If we measure the first register, we obtain b such that $|\frac{b}{Q} - \frac{c}{r}| \leq \frac{1}{2Q}$ with high probability for some random c. If this is the case, $\frac{c}{r}$ is the only fraction with $c \wedge r = 1$ and $r \leq N$ such that $|\frac{b}{Q} - \frac{c}{r}| \leq \frac{1}{2Q}$ (proof omitted). This is because we chose Q such that $Q \geq N^2$.

If we indeed have $c \wedge r = 1$, which still happens with probability greater than $\Omega(\frac{1}{\log(\log(r))})$, we can use the continuous fraction method to find the unique fraction $\frac{c}{r}$ satisfying $|\frac{b}{Q} - \frac{c}{r}| \leq \frac{1}{2Q}$ from which we can get r.

DONE :)

Complexity of the period finding algorithm

We apply the above procedure until we find c such that $c \wedge r = 1$. This means we perform $O(\log(\log(r)))$ loops. Each loop makes 2 calls to QFT_Q and 1 call to O_f.

The running time is therefore $PFA_f \leq O\left(\log(\log(r))(QFT_Q + O_f)\right)$.

2.2.2 Complexity of Shor's algorithm

Shor's algorithm: $O(1)$ calls to PFA_f with $f(k) = x^k \mod N$ for some random x.

One can show that O_f can be calculated in $O(\log^2(N)\log(\log(N))\log(\log(\log(N))))$ using efficient squaring. (Perfect) QFT_Q is made in $O(Q^2) = O(\log^2(N))$.

From there, we conclude that the total running time of Shor's algorithm is

$$\begin{aligned} O\left(\log(\log(r)) * \left(\log^2(N) + \log^2(N)\log(\log(N))\log(\log(\log(N)))\right)\right) &\leq O\left(\log(n) * \left(n^2 + n^2\log(n)\log(\log(n))\right)\right) \\ &= O(n^2 polylog(n)). \end{aligned}$$

Chapter 3

Quantum random walks

3.1 Classical random walks

3.1.1 Introduction

Consider an undirected graph $G = (V, E)$ with N vertices. Suppose at least an ε-fraction of the vertices are marked.
We consider the following algorithm: start from a specific vertex y

- If y is marked → OK.
- Otherwise: go to a random neighbor of y and continue the random walk.

It is a naive algorithm for finding a marked element. One advantage however is that the space required is $O(\log(N))$. The algorithm can be useful depending on the data structure used for encoding the graph.

3.1.2 Analysis of the classical random walk

Here, we only consider d-regular graphs without self loops, *i.e.* each vertex has *exactly* d neighbors.
For such a graph G, we consider the matrix $P = \frac{1}{d}Adj(G)$, where $Adj(G)$ is the adjacency matrix of G. By definition, we have

$$P_{xy} = \frac{1}{d} \text{ if } (x,y) \in E \quad ; \quad P_{xy} = 0 \text{ otherwise.}$$

When performing a random walk on graph, we keep track of the probability of being in each vertex using a column vector $v = \begin{pmatrix} v_0 \\ \vdots \\ v_{N-1} \end{pmatrix}$ satisfying $\sum_{i=0}^{N-1} v_i = 1$.

If we start from a distribution v, $P \cdot v$ will correspond to the new distribution on vectors after applying 1 step of the random walk.
For example, we initially start with the vector v^0 satisfying $v^0_y = 1$ and $v^0_x = 0$ for $x \neq y$. After 1 step of the walk, the new distribution $v^1 = Pv^0$ satisfies $v^1_x = \frac{1}{d}$ if $(x,y) \in E$ and

$v_x^1 = 0$ otherwise. Similarly, after k steps of the random walk, we are in the distribution $v^k = P^k v^0$.

Convergence speed

Starting from any probability distribution vector (pdv) v, how fast do we converge to the uniform distribution $u = \begin{pmatrix} \frac{1}{N} \\ \vdots \\ \frac{1}{N} \end{pmatrix}$?

We have to look at the eigenvalues of P. P is a real symmetric matrix. Let $\lambda_1 \geq \lambda_2 \geq \dots \lambda_N$ the N (not necessarily distinct) eigenvalues of P. Since $Pu = u$, we have $\lambda_1 = 1$. Moreover, a quick look at P tells us that $\forall i \in \{2, \dots, N\}, \lambda_i \in]-1, 1[$. Moreover, each corresponding normalized eigenstate v_i for $i \in \{2, \dots, N\}$ is orthogonal to u.

Let $\delta := \lambda_1 - \max_{i \in \{2,\dots,N\}} |\lambda_i| = 1 - \max_{i \in \{2,\dots,N\}} |\lambda_i|$ the spectral gap of P. δ will determine at what rate any probability distribution vector v will converge to u. For any pdv v, we write $v = \sum_i \alpha_i v_i$ and

$$\begin{aligned} P^k v &= \sum_{i=1}^{N} \alpha_i \lambda_i^k v_i \\ &= u + \sum_{i=2}^{N} \alpha_i \lambda_i^k v_i \end{aligned}$$

which gives

$$\left|\left|P^k(v) - u\right|\right|^2 = \left|\left|\sum_{i=2}^{N} \alpha_i \lambda_i^k v_i\right|\right| = \sum_{i=2}^{N} |\alpha_i|^2 |\lambda_i|^{2k} \leq (1-\delta)^{2k} ||v||^2 \leq (1-\delta)^{2k}.$$

If δ is not too small, the convergence rate is very fast. If we take $k = \ln(\frac{1}{\eta})\frac{1}{\delta}$, we have

$$\left|\left|P^k(v) - u\right|\right|^2 \leq \eta.$$

Once we are close to the uniform distribution u, we have an ε chance of hitting a marked vertex. If we miss, repeat.

Cost of the random walk:

- S (setup): cost to set up an initial v.
- U (update): cost to perform one step of the random walk.
- C (check): cost to check whether a vertex is marked.

Consider a classical search algorithm that starts at v, and then repeats the following until it finds a marked vertex: check if the current vertex is marked, and if not run a random walk for roughly $\frac{1}{\delta}$ steps to get close to the uniform distribution. Ignoring constant factors, the expected cost before this procedure finds a marked item, is on the order of

$$S + \frac{1}{\varepsilon}\left(C + \frac{1}{\delta}U\right).$$

3.2 Quantum random walks

We will try to mimic random walks. In a quantum random walk, we will not only keep track of the current vertex but also of the neighbors (or predecessors). Quantum states: superposition of elements $|x\rangle |y\rangle$ where x is the current vertex and y a neighbor of x. Instead of looking at the classical probability vector of the current position, we will look at its quantum superposition.

Quantum walk on d-regular graph

- **Input:** A graph $G = (V, E)$ which is connected and d-regular. A set of marked elements $M \subseteq V$ with $|M| = \varepsilon|V|$.
- **Goal**: Find $x \in M$.

A quantum walk will be similar to Grover's algorithm:

- For all $x \in V$, Let $|p_x\rangle = \sum_{y \in V} \sqrt{P_{xy}} |y\rangle = \frac{1}{\sqrt{d}} \sum_{y:(x,y) \in E} |y\rangle$.
- Let $|U\rangle = \frac{1}{\sqrt{N}} \sum_{x \in V} |x\rangle |p_x\rangle$. $|U\rangle$ = uniform superposition of vertices x with uniformly associated neighbors.
- As in Grover, we decompose into the good state (meaning where x is marked) and the bad state. Let M be the set of marked vertices.

  $$\text{We write} \quad |U\rangle = \sqrt{\frac{|M|}{N}} |G\rangle + \sqrt{\frac{N - |M|}{N}} |B\rangle$$

  with $|G\rangle = \frac{1}{\sqrt{|M|}} \sum_{x \in M} |x\rangle |p_x\rangle$ and $|B\rangle = \frac{1}{\sqrt{N - |M|}} \sum_{x \notin M} |x\rangle |p_x\rangle$.
- We can perform the same reflexions as in Grover's algorithm to get close to $|G\rangle$.

A quantum walk algorithm therefore performs the following:

Quantum algorithm for finding a marked element

1. **Setup**: construct the state $|U\rangle = \sqrt{\frac{|M|}{N}} |G\rangle + \sqrt{\frac{N-|M|}{N}} |B\rangle$.
2. We perform the following $O(\frac{1}{\sqrt{\varepsilon}}) = O(\sqrt{\frac{N}{|M|}})$ times
   - Do a reflexion over $|B\rangle$ (in the $\{|B\rangle, |G\rangle\}$ subspace).
   - Do a reflexion of $|U\rangle$.
3. Measure the first register to see if it is a marked state.
4. The above algorithm will find a marked element with high probability.

How to construct those 2 reflexions?

Reflexion over $|B\rangle$:

- The unitary O_Z that performs $|x\rangle |y\rangle \rightarrow (-1)^{x\in M} |x\rangle |y\rangle$ will do the job.
- One can check that $O_Z(|B\rangle) = |B\rangle$ and $O_Z(|G\rangle) = -|G\rangle$.
- This requires a quantum algorithm that checks whether an element is marked or not: Quantum **Check** procedure.

Reflexion over $|U\rangle$: This is the part where the quantum walk framework will give an advantage. In Grover's algorithm, we performed this reflexion over $|U\rangle$ using a circuit that constructs $|U\rangle$. This is can be done with the Setup procedure. We will show however that there is a more efficient way of performing this reflexion, and this justifies the whole quantum walk framework.

3.2.1 Reflexion over $|U\rangle$

We have

$$|U\rangle = \frac{1}{\sqrt{N}} \sum_{x\in V} |x\rangle |p_x\rangle = \frac{1}{\sqrt{N}} \sum_{x\in V} |x\rangle \sum_{y:(x,y)\in E} \frac{1}{\sqrt{d}} |y\rangle = \frac{1}{\sqrt{N}} \sum_{y\in V} |p_y\rangle |y\rangle .$$

To perform a reflexion of $|U\rangle$, we first present a unitary $W(P)$ for which the only eigenvector with eigenvalue 1 is $|U\rangle$.

Construction of $W(P)$

- Let $\mathcal{A} = span\{|x\rangle |p_x\rangle\}$ and $\mathcal{B} = span\{|p_y\rangle |y\rangle\}$. Let $Ref(\mathcal{A})$ be the reflexion through $\mathcal{A}$ *i.e.* $\forall |\psi\rangle \in \mathcal{A}, Ref(\mathcal{A})(|\psi\rangle) = |\psi\rangle$ and $\forall |\psi\rangle \notin \mathcal{A}, Ref(\mathcal{A})(|\psi\rangle) = -|\psi\rangle$. Similarly, we define $Ref(\mathcal{B})$.
- $W(P) = Ref(\mathcal{B})Ref(\mathcal{A})$.

We have $|U\rangle \in \mathcal{A}$ and $|U\rangle \in \mathcal{B}$ so $W(P)|U\rangle = |U\rangle$. So we constructed a quantum unitary $W(P)$ such that $|U\rangle$ is an eigenstate of $W(P)$ with eigenvalue 1. Recall $P = \frac{1}{d}Adj(G)$. Let $\lambda_1 = 1 > \lambda_2 > \cdots \geq \lambda_N > -1$ the N ordered eigenvalues of P. Let θ_i such that $\lambda_i = \cos(\theta_i)$ ($\theta_1 = 0$).

Theorem 3.1 (Admitted)**.** *The eigenvalues of $W(P)$ are of the form $e^{\pm 2i\theta_j}$. Eigenvalue 1 has a unique eigenvector, and it is $|U\rangle$. Recall also the definition of the spectral gap δ: $\max_{i\geq 2} |\lambda_i| \leq 1-\delta$. For all $j \geq 2$, we also have $|\theta_j| \geq \sqrt{2\delta}$.*

We use phase estimation (PE) now to perform the reflexion over $|U\rangle$. We have a unitary $W(P)$ with eigenvectors $e^{\pm 2i\theta_j}$. Eigenvalue 1 ($\theta_1 = 0$) has a unique eigenvector $|U\rangle$ and for $j \geq 2$, $|\theta_j| \geq \sqrt{2\delta}$.

Reflexion over $|U\rangle$ using PE

1. Start from an eigenvector $|\Psi\rangle|0\rangle$ of U with eigenvalue $e^{2\pi i\theta}$.
2. Apply PE with precision $\frac{\sqrt{\delta}}{2}$ to get $|\Psi\rangle\left|\widetilde{\theta}\right\rangle$ with $|\widetilde{\theta}-\theta| < \frac{\sqrt{\delta}}{2}$.
3. Apply Z_2 satisfying $Z_2(|\Psi\rangle\left|\widetilde{\theta}\right\rangle) = (-1)^{|\widetilde{\theta}|>\frac{\sqrt{\delta}}{2}}|\Psi\rangle\left|\widetilde{\theta}\right\rangle$.
4. Apply PE^{-1} to get $(-1)^{|\widetilde{\theta}|>\frac{\sqrt{\delta}}{2}}|\Psi\rangle|0\rangle$

Analysis of the unitary:

- If $|\Psi\rangle = |U\rangle$ then $\theta = 0$ and $\widetilde{\theta} < \frac{\sqrt{\delta}}{2}$ so $R(P)(|\Psi\rangle|0\rangle) = |\Psi\rangle|0\rangle$.
- If $|\Psi\rangle$ is any other eigenvector, $|\theta| \geq \sqrt{2\delta}$ and $|\widetilde{\theta}| \geq \sqrt{2\delta} - \frac{\sqrt{\delta}}{2} > \frac{\sqrt{\delta}}{2}$ so $R(P)(|\Psi\rangle|0\rangle) = -|\Psi\rangle|0\rangle$. By linearity, this extends to any state orthogonal to $|U\rangle$.
- The above unitary is exactly a reflexion over $|U\rangle$! Uses $O(\frac{1}{\sqrt{\delta}})$ calls to $W(P)$.

3.2.2 Recap of the random walk

Start from a graph $G = (V, E)$ which is connected and d-regular with an ε-fraction of marked vertices. Goal: find a marked vertex.

Quantum algorithm for finding a marked element

1. **Setup**: construct the state $|U\rangle = \sqrt{\frac{|M|}{N}}|G\rangle + \sqrt{\frac{N-|M|}{N}}|B\rangle$.
2. We perform the following $O(\frac{1}{\sqrt{\varepsilon}}) = O(\sqrt{\frac{N}{|M|}})$ times
    - Do a reflexion over $|B\rangle$: uses 1 **Check** procedure.
    - Do a reflexion of $|U\rangle$: uses $O(\frac{1}{\sqrt{\delta}})$ **Update** procedures.
3. Measure the first register to see if it is a marked state.
4. The above algorithm will find a marked element with high probability.

$$\textbf{Total cost} = S + O(\frac{1}{\sqrt{\varepsilon}})\left(C + O(\frac{1}{\sqrt{\delta}})U\right). \tag{3.1}$$

Compare with classical cost $S + O(\frac{1}{\varepsilon})\left(C + O(\frac{1}{\delta})U\right)$.

About the Update procedure: So the update procedure consists of constructing the unitary $W(P) = Ref(B)Ref(A)$. Suppose we are able to implement the following two operations

1. $|x\rangle|0\rangle \to |x\rangle|p_x\rangle$.
2. $|0\rangle|y\rangle \to |p_y\rangle|y\rangle$.

Since (1) and (2) prepare a uniform superposition over the neighbors of x and y, respectively, one can think of them as taking one classical walk step "in superposition." Note that $ref(A)$ can be implemented by applying the inverse of (1), putting a minus if the second register is not $|0\rangle$, and applying (1). We can similarly implement ref(B) using (2) and its inverse. Hence we can think of $W(P) = ref(B)ref(A)$ as corresponding to four steps of the classical walk in superposition.

In practice, these operations can be constructed from an operation that construct from a node one of its neighbors (and then can be used in superposition over the possible neighbors).

3.3 Application: Golden collision finding

We consider an efficiently computable function $f : \{0,1\}^n \to \{0,1\}^n$ st. there exists a unique pairs $x_0, x_1 \neq x_0$ st. $f(x_0) = f(x_1)$. We already saw the BHT algorithm that succeeds wp. $2^{n/3}$ but when we have many (typically $O(2^n)$) collisions. Here, we only have a single solution so the BHT algorithm won't work. A priori, we don't know how to do better than Grover search on all the input pairs (x_0, x_1) which takes time $O(\sqrt{2^{2n}}) = O(2^n)$.

Here, we will show how to use a quantum walk to solve this problem in time $O(2^{2n/3})$. In order to do this, we have to construct a graph G on which we perform the random walk. The graph $G = (V, E)$ is the following, for a parameter r:

- For $S \subseteq \{0,1\}^n$, we define $v_S = (v_S(inp), v_S(out))$ where $v_S(inp) = \{x_i\}_{i\in S}$ and $v_S(out) = \{f(x_i)\}_{i\in S}$. We have $V = \{v_S : |S| = r\}$.
- $(v_S, v_{S'})$ form an edge iff. S and S' differ in exactly one position, so there exists $y \in S(inp)$ and $y' \in S'(inp)\backslash S(inp)$ st. $S'(inp) = (S(inp)\backslash\{y\}) \cup \{y'\}$.
- A vertex v is marked iff. the golden collision (x_0, x_1) appears in v, so $x_0 \in v(inp)$ and $x_1 \in v(inp)$.

Here, we are only interested in the query complexity of the algorithm, so only the number of calls to O_f It is also possible to perform a real time analysis of this protocol but it is much more cumbersome. G is a Johnson graph $J(2^n, r)$, so it has a spectral gap $\delta = \frac{2^n}{r(2^n - r)} \approx \frac{1}{r}$ for $r \ll 2^n$. Now, let's calculate each term of Equation 3.1. We have

- *Setup* $= r$. Constructing a vertex requires r queries and this can be done in superposition, so we can construct $|U\rangle$ with r queries.
- *Update* $= O(1)$. In order to go from a vertex v_S to a vertex $v_{S'}$, with $S'(inp) = (S(inp)\backslash\{y\}) \cup \{y'\}$, we have to uncompute the value $f(y)$ and add the value $f(y')$ which requires 2 queries to O_f. Then, we can apply this in superposition in order to construct the unitary $|v\rangle|0\rangle \to |v\rangle|p_v\rangle$, and similarly for the other unitary.
- The checking cost is 0, all the information is in $v_S(out)$.

- The fraction of marked vertices is $\varepsilon = \frac{r^2}{2^{2n}}$

Putting everything together, we have

$$Cost = S + O(\frac{1}{\sqrt{\varepsilon}}) \left(C + O(\frac{1}{\sqrt{\delta}})U \right) = r + \frac{n}{r} \left(\sqrt{r} + 0 \right) = O(r + \frac{2^n}{\sqrt{r}}).$$

By taking $r = 2^{2n/3}$, we obtain the desired result.

Chapter 4

General formalism of quantum computing

4.1 Mixed states

Consider the state $|\Phi^+\rangle_{AB} = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ shared between 2 parties *Alice* and *Bob*. How do we describe Alice and Bob's state locally? Assume Alice measures her state in the computational basis. From the laws of partial measurement she measures "0" wp. $\frac{1}{2}$ and Bob has the state $|0\rangle$; or she measures "1" wp. $\frac{1}{2}$ and Bob has the state $|1\rangle$. If we look only from Bob's perspective, and if we define ρ_B his state, we have $\rho_B = \begin{cases} |0\rangle \text{ wp. } \frac{1}{2} \\ |1\rangle \text{ wp. } \frac{1}{2} \end{cases}$.

A mixed state (or density matrix) is a clean way of describing probabilistic quantum states as the one described above. A state

$$\rho = \begin{cases} |e_1\rangle \text{ wp. } p_1 \\ \vdots \\ |e_k\rangle \text{ wp. } p_k \end{cases}$$

is written $\rho = \sum_i p_i |e_i\rangle\langle e_i|$. If these states are n qubit states, recall Dirac's notation: $|e_i\rangle$ is a column vector of and $\langle e_i|$ is a line vector and $|e_i\rangle\langle e_i|$ is the multiplication of the two which gives a matrix. For example: $|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$, then

$$|\psi\rangle\langle\psi| = \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \cdot \begin{pmatrix} \alpha^* & \beta^* \end{pmatrix} = \begin{pmatrix} \alpha\alpha^* & \alpha\beta^* \\ \beta\alpha^* & \beta\beta^* \end{pmatrix}.$$

A few notable examples on 1 qubit:

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

Definition 4.1. *A mixed state on n qubits is a matrix $\rho = \sum_i p_i |e_i\rangle\langle e_i|$ where each $|e_i\rangle$ is an n-qubit state, each $p_i \geq 0$ and $\sum_i p_i = 1$.*

Notice that the states $|e_i\rangle$ in the decomposition of a mixed state needn't be orthogonal. For example:

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

is a valid 1-qubit mixed state.

Properties of quantum mixed states.

- A quantum mixed state ρ is a Hermitian matrix $\rho = \rho^* := \overline{\rho}^\intercal$. This is because each $|\phi\rangle\langle\phi|$ is Hermitian.
- $\mathrm{Tr}(\rho) = 1$ since $\mathrm{Tr}(|\phi\rangle\langle\phi|) = 1$ for each $|\phi\rangle$.
- Since ρ is Hermitian it is diagonalizable with real valued eigenvalues, and moreover, these eigenvalues are non-negative (since the $p_i \geq 0$ in the definition). This means we can write $\rho = \sum_i \lambda_i |e_i\rangle\langle e_i|$ $0 \leq \lambda_i \leq 1$, $\sum_i \lambda_i = 1$ and the $|e_i\rangle$ are pairwise orthogonal quantum states.

4.1.1 Applying quantum operations on mixed states

A mixed state $\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i|$ is a complete description of the quantum state you have.

Unitaries. Applying a unitary U on a pure state $|\psi\rangle$ gives the state $U|\psi\rangle = |\phi\rangle$. If we from a density matrix $\rho = |\psi\rangle\langle\psi|$, then applying a unitary U on this state gives the state $|\phi\rangle\langle\phi| = U|\psi\rangle\langle\psi|U^\dagger$. More generally, applying U on a state ρ gives the state $U\rho U^\dagger$.

Projective measurements. Consider a state ρ of n qubits and a basis $B = |b_1\rangle, \ldots, |b_{2^n}\rangle$ of the Hilbert space of n qubits. If you measure ρ in the basis B, you have

$$\Pr[\text{ outcome } |b_k\rangle] = \sum_i p_i |\langle\psi_i|b_k\rangle|^2 = \langle b_k|\rho|b_k\rangle.$$

4.1.2 Different mixtures of quantum states can have the same density matrix

Let $\rho_1 = \frac{3}{4}|0\rangle + \frac{1}{4}|1\rangle$ and $\rho_2 = \frac{1}{2}|0\rangle\langle 0| + \frac{1}{4}|+\rangle\langle +| + \frac{1}{4}|-\rangle\langle -|$. We have

$$\rho_1 = \begin{pmatrix} 3/4 & 0 \\ 0 & 1/4 \end{pmatrix}$$

and

$$\rho_2 = \begin{pmatrix} 1/2 & 0 \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} 1/8 & 1/8 \\ 1/8 & 1/8 \end{pmatrix} + \begin{pmatrix} 1/8 & -1/8 \\ -1/8 & 1/8 \end{pmatrix} = \begin{pmatrix} 3/4 & 0 \\ 0 & 1/4 \end{pmatrix}.$$

Two different decompositions can lead to the same density matrix. This means that these two quantum states are the same, they can't be distinguished one from the other by using quantum operations.

4.2 Partial trace

Let's go back to our original motivation. Assume you have a state $|\psi\rangle_{AB} = \sum_{i,j} \alpha_i |e_i\rangle_A |i\rangle_B$ shared between 2 parties Alice and Bob. What is the state that Alice has? She has the mixed state

$$\rho_A = \sum_i |\alpha_i|^2 |e_i\rangle\langle e_i|.$$

To see this, assume Bob measures his state in the computational basis. He gets outcome "i" wp. $|\alpha_i|^2$ and Alice has the state $|e_i\rangle$ which means she has the state ρ_A. Notice also that Alice's state doesn't depend on whether Bob has measured or not, so we can always describe Alice's state as ρ_A.

The mathematical operation that describes Alice's reduced state is called the *partial trace*. For a (possibly mixed) state ρ_{AB} shared between Alice and Bob, we define

$$\mathrm{Tr}_B(\rho_{AB}) = \sum_j (I_A \otimes \langle j|)\rho_{AB}(I_A \otimes |j\rangle). \tag{4.1}$$

$\mathrm{Tr}_B(\rho_{AB})$ means that we "trace out" Bob's registers from ρ_{AB} (so we keep Alice's part). We will rarely use Equation 4.1 directly. Rather, we will use the following results:

- For $|\psi\rangle_{AB} = \sum_{i,j} \alpha_i |e_i\rangle_A |i\rangle_B$, $\mathrm{Tr}_B(|\psi_{AB}\rangle\langle\psi_{AB}|) = \sum_i |\alpha_i|^2 |e_i\rangle\langle e_i|$.
- For $\rho_{AB} = \sum_i p_i |f_i\rangle\langle f_i|$, $\mathrm{Tr}_B(\rho_{AB}) = \sum_i p_i \mathrm{Tr}_B |f_i\rangle\langle f_i|$.

With the partial, we are now able to characterize the reduced state of a quantum state share between different registers.

4.3 Generalized measurements

POVM, for Positive Operator Value Measurements, generalize projective measurements.

Definition 4.2. *A POVM is an ensemble of matrices $\{M_i\}_i$ st. $\sum_i M_i M_i^\dagger = I$. Measuring a state ρ with this POVM gives outcome i wp. $p_i = tr(\rho M_i M_i^\dagger)$ and conditioned on obtaining outcome i, the resulting state is*

$$\rho_i = \frac{M_i \rho M_i^\dagger}{tr(M_i \rho M_i^\dagger)}.$$

Remarks.

- A POVM is sometimes defined by the matrices $F_i = M_i M_i^\dagger$. Be careful however, the probabilities p_i depend only on F_i but the resulting states ρ_i actually depend on the M_i so using the F_i is fine if you are only interested in the outcome distribution but you need the M_i is you want to specify the resulting states.
- There is no restriction on the M_i but the $F_i = M_i M_i^\dagger$ are positive semi-definite (hence the name POVM), meaning that for any pure state $|\psi\rangle$,

$$\langle\psi| F_i |\psi\rangle \geq 0.$$

Moreover, any positive semi-definite matrix F_i is of the form $M_i M_i^\dagger$ so we can describe the POVM $\{F_i\}$ with $\sum_i F_i = I$ if we are only interested in the outcome probabilities.

- Projective measurements are a special case, where the M_i are projectors (which implies $M_i = M_i M_i^\dagger = F_i$).
- Physically, a POVM on a state ρ corresponds to the setting where we add some extra qubits $|0^m\rangle\langle 0^m|$ to ρ, perform a projective measurement and then trace-out some qubits. So POVM are not more powerful from a physical point of view but are an elegant and compact form for describing these operations.

4.4 Purifications

A purification $|\psi\rangle_{AB}$ of a state ρ_B satisfies $\mathrm{Tr}_A|\psi_{AB}\rangle\langle\psi_{AB}| = \rho_B$. For example, if $\rho_B = \sum_i p_i|f_i\rangle\langle f_i|$ then the state $|\phi\rangle_{AB} = \sum_i \sqrt{p_i}\,|i\rangle_A\,|f_i\rangle_B$ is a purification of ρ_B.

Proposition 4.3 (Schmidt Decomposition)**.** *Let $|\psi\rangle_{AB}$ be a state of $2n$ qubits, where each register A, B contains n qubits. There exists two basis $\{|e_1\rangle, \ldots, |e_{2^n}\rangle\}$ and $\{|f_1\rangle, \ldots, |f_{2^n}\rangle\}$ st. $|\psi\rangle_{AB} = \sum_{i=1}^{2^n} \alpha_i\,|e_i\rangle_A\,|f_i\rangle_B$. with $\sum_i |\alpha_i|^2 = 1$. This decomposition is unique. Moreover,*

$$\mathrm{Tr}_A|\psi_{AB}\rangle\langle\psi_{AB}| = \sum_i |\alpha_i|^2|f_i\rangle\langle f_i| \ ; \ \mathrm{Tr}_B|\psi_{AB}\rangle\langle\psi_{AB}| = \sum_i |\alpha_i|^2|e_i\rangle\langle e_i|.$$

Proposition 4.4. *Assume we have two quantum pure states $|\phi_{AB}\rangle$ and $|\psi_{AB}\rangle$ st. $\mathrm{Tr}_A(|\phi_{AB}\rangle\langle\phi_{AB}|) = \mathrm{Tr}_A(|\psi_{AB}\rangle\langle\psi_{AB}|) = \rho_B$. There exists a unitary U acting on A st. $(U \otimes I)\,|\phi_{AB}\rangle = |\psi_{AB}\rangle$.*

Proof. We write $\rho_B = \sum_i p_i|f_i\rangle\langle f_i|$ the spectral decomposition of ρ_B (so all the $|f_i\rangle$ are pairwise orthogonal). This means we can write $|\phi_{AB}\rangle$ and $|\psi_{AB}\rangle$ as follows, using the Schmidt decomposition.

$$|\phi_{AB}\rangle = \sum_i \alpha_i\,|e_i\rangle\,|f_i\rangle$$

$$|\phi_{AB}\rangle = \sum_i \alpha'_i\,|e'_i\rangle\,|f_i\rangle$$

with $|\alpha_i| = |\alpha'_i| = \sqrt{p_i}$ and $\{|e_i\rangle\}$ as well as the $\{|e'_i\rangle\}$ each form a basis. This means there exists a unitary U st. for each i, $U\,|e_i\rangle = \frac{\alpha'_i}{\alpha_i}\,|e'_i\rangle$. We then immediately have

$$(U \otimes I)\,|\phi_{AB}\rangle = |\psi_{AB}\rangle\,.$$

□

Chapter 5

Distance measures for quantum states and first notion of quantum information theory

5.1 How close are two quantum states?

5.1.1 The trace distance

We introduce here the notion of trace distance, which is very useful in determining how close two mixed states are. We present here basic properties of this distance. More about the trace distance can be found in [NC00].

Definition and basic properties

Definition 5.1. *For any two quantum mixed states ρ and σ, the trace distance between ρ and σ is defined as $\Delta(\rho,\sigma) = \frac{1}{2}\|\rho-\sigma\|_{\mathrm{tr}}$ where $\|M\|_{\mathrm{tr}} = Tr(\sqrt{M^\dagger M})$.*

Since ρ and σ are hermitian, we have

$$\Delta(\rho,\sigma) = \frac{1}{2}Tr(\sqrt{(\rho-\sigma)^\dagger(\rho-\sigma)}).$$

Be careful, this *doesn't necessarily imply* $\Delta(\rho,\sigma) = \frac{1}{2}Tr(\rho-\sigma)$!

$\rho-\sigma$ is Hermitian but not necessarily positive. This means we can write $\rho-\sigma = \sum_i \lambda_i |e_i\rangle\langle e_i|$ where $\{|e_i\rangle\}_i$ is a orthonormal basis and the $\lambda_i \in \mathbb{R}$. We have $\Delta(\rho,\sigma) = \frac{1}{2}\sum_i |\lambda_i|$.

Notice also that $\sum_i \lambda_i = Tr(\rho-\sigma) = Tr(\rho) - Tr(\sigma) = 1 - 1 = 0$.

The trace distance is a distance. Indeed, it satisfies the following properties:

- $\Delta(\rho,\sigma) = 0 \Leftrightarrow \rho = \sigma$.

- $0 \leq \Delta(\rho, \sigma) \leq 1$.
- $\Delta(\rho, \sigma) = \Delta(\sigma, \rho)$.
- $\forall \rho, \sigma, \tau,\ \Delta(\rho, \tau) \leq \Delta(\rho, \sigma) + \Delta(\sigma, \tau)$

Example of Trace distances

- ρ and σ are diagonalizable in the same basis : this means we can write $\rho = \sum_i p_i|e_i\rangle\langle e_i|$ and $\sigma = \sum_i q_i|e_i\rangle\langle e_i|$ where $\{|e_i\rangle\}_i$ is a orthonormal basis. In this case, we have $\Delta(\rho, \sigma) = \frac{1}{2}\sum_i |p_i - q_i|$.
- ρ and σ are two pure states : this means we can write $\rho = |\psi\rangle\langle\psi|$ and $\sigma = |\phi\rangle\langle\phi|$. In this case, we have $\Delta(\rho, \sigma) = \sqrt{1 - |\langle\psi|\phi\rangle|^2}$.
- Other example: $\rho = |0\rangle\langle 0|$, $\sigma = \frac{3}{4}|+\rangle\langle+| + \frac{1}{4}|-\rangle\langle-|$. Let's calculate $\Delta(\rho, \sigma)$ using the definition. We have

$$\begin{aligned}\rho - \sigma &= \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} - \left[\frac{3}{4}\begin{pmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{pmatrix} + \frac{1}{4}\begin{pmatrix} 1/2 & -1/2 \\ -1/2 & 1/2 \end{pmatrix}\right] \\ &= \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} - \begin{pmatrix} 1/2 & 1/4 \\ 1/4 & 1/2 \end{pmatrix} \\ &= \begin{pmatrix} 1/2 & -1/4 \\ -1/4 & -1/2 \end{pmatrix}\end{aligned}$$

Calculation tip: at that point, make sure that $\rho - \sigma$ is Hermitian and that its trace is 0.

$$(\rho - \sigma)^\dagger(\rho - \sigma) = \begin{pmatrix} 1/2 & -1/4 \\ -1/4 & -1/2 \end{pmatrix} \cdot \begin{pmatrix} 1/2 & -1/4 \\ -1/4 & -1/2 \end{pmatrix} = \begin{pmatrix} 5/16 & 0 \\ 0 & 5/16 \end{pmatrix}$$

From there, we have

$$\sqrt{(\rho - \sigma)^2} = \sqrt{\begin{pmatrix} 5/16 & 0 \\ 0 & 5/16 \end{pmatrix}} = \begin{pmatrix} \sqrt{5}/4 & 0 \\ 0 & \sqrt{5}/4 \end{pmatrix}$$

which allows us to conclude that $\Delta(\rho, \sigma) = \frac{1}{2}Tr(\sqrt{(\rho - \sigma)(\rho - \sigma)^\dagger}) = \sqrt{5}/4$.

Invariance over unitary operations. The trace distance has the following property:

Proposition 5.2. *For any two quantum mixed states ρ, σ and any unitary operation U, we have $\Delta(\rho, \sigma) = \Delta(U\rho U^\dagger, U\sigma U^\dagger)$.*

Interpretation of the trace distance

Let's consider two people Alice and Bob. Alice has bit b unknown to Bob. Suppose now Alice sends a mixed state ρ_b that depends on b. With what probability can Bob guess b? This probability in fully characterized by the trace distance between ρ_0 and ρ_1. We have:

Proposition 5.3. $\max(\Pr[\textit{Bob guesses } b]) = \frac{1}{2} + \frac{\Delta(\rho_0, \rho_1)}{2}$

Proof. We will not go through the whole proof. However, we'll show that max(Pr[Bob guesses b]) $\geq \frac{1}{2} + \frac{\Delta(\rho_0,\rho_1)}{2}$. To do this, we present a measurement for Bob that allows him to guess b with probability $\frac{1}{2} + \frac{\Delta(\rho_0,\rho_1)}{2}$.

We write $\rho_0 - \rho_1 = \sum_i \lambda_i |e_i\rangle\langle e_i|$ where $\{|e_i\rangle\}_i$ is a orthonormal basis and $\sum_i \lambda_i = 0$. Bob's strategy is to measure in the $\{|e_1\rangle, \ldots, |e_n\rangle\}$ basis. Suppose Bob's outcome is $|e_i\rangle$:

- If $\lambda_i \geq 0$, Bob's guess is 0.
- If $\lambda_i < 0$, Bob's guess is 1.

Let

$$p_i = \langle e_i|\rho_0|e_i\rangle = \Pr[\text{Bob's outcome is } |e_i\rangle \mid \text{Bob receives } \rho_0]$$
$$q_i = \langle e_i|\rho_1|e_i\rangle = \Pr[\text{Bob's outcome is } |e_i\rangle \mid \text{Bob receives } \rho_1]$$

Notice that we have $\langle e_i|\rho_0 - \rho_1|e_i\rangle = \lambda_i = p_i - q_i$. We write

$$\Pr[\text{Bob guesses } b \text{ correctly } |b = 0] = \sum_{i:\lambda_i \geq 0} p_i$$
$$\Pr[\text{Bob guesses } b \text{ correctly } |b = 1] = \sum_{i:\lambda_i < 0} q_i$$

Since b is a random bit, we have

$$\Pr[\text{Bob guesses } b \text{ correctly}] = \frac{1}{2}\sum_{i:\lambda_i \geq 0} p_i + \frac{1}{2}\sum_{i:\lambda_i < 0} q_i$$

Moreover, we have

$$\begin{aligned}\sum_i |\lambda_i| = \sum_i |p_i - q_i| &= \sum_{i:\lambda_i \geq 0} p_i - q_i + \sum_{i:\lambda_i < 0} q_i - p_i \\ &= \sum_{i:\lambda_i \geq 0} p_i - (1 - \sum_{i:\lambda_i < 0} q_i) + \sum_{i:\lambda_i < 0} q_i - (1 - \sum_{i:\lambda_i \geq 0} p_i) \quad \textit{using} \sum_i p_i = \sum_i q_i = 1 \\ &= 2(\sum_{i:\lambda_i \geq 0} p_i) + 2(\sum_{i:\lambda_i < 0} q_i) - 2\end{aligned}$$

From there, we conclude:

$$\Pr[\text{Bob guesses } b \text{ correctly}] = \frac{1}{2}\sum_{i:\lambda_i \geq 0} p_i + \frac{1}{2}\sum_{i:\lambda_i < 0} q_i = \frac{1}{4}\sum_i |\lambda_i| + \frac{1}{2} = \frac{\Delta(\rho,\sigma)}{2} + \frac{1}{2}$$

NB: This measurement is optimal for Bob □

5.2 Fidelity for quantum states

We now present a second notion for quantifying how close two quantum states are, *the fidelity.* We will use this notion to analyze more formally cheating possibilities in quantum bit commitment protocols.

5.2.1 Definition and basis properties

Definition 5.4. *For any two quantum mixed states ρ and σ, the fidelity between ρ and σ is defined as $F(\rho, \sigma) = Tr(\sqrt{\sqrt{\rho}\sigma\sqrt{\rho}})$*

The fidelity has the following properties

- $0 \leq F(\rho, \sigma) \leq 1$.
- $F(\rho, \sigma) = 1 \Leftrightarrow \rho = \sigma$
- $F(\rho, \sigma)$

It seems that the quantity $(1 - F(\rho, \sigma)$ has similar properties than $\Delta(\rho, \sigma)$. However, the quantity $1 - F$ does not satisfy the triangle inequality, meaning we don't necessarily have

$$(1 - F(\rho, \tau)) \leq (1 - F(\rho, \sigma)) + (1 - F(\sigma, \tau))$$

. However, we have a 'weak' triangle inequality in the following form

Proposition 5.5. *For any states ρ, σ, τ, we have $(1 - F(\rho, \tau)) \leq 2(1 - F(\rho, \sigma)) + 2(1 - F(\sigma, \tau))$*

Example of fidelities

- ρ and σ are diagonalizable in the same basis : this means we can write $\rho = \sum_i p_i |e_i\rangle\langle e_i|$ and $\sigma = \sum_i q_i |e_i\rangle\langle e_i|$ where $\{|e_i\rangle\}_i$ is a orthonormal basis. In this case, we have $F(\rho, \sigma) = \sum_i \sqrt{p_i q_i}$.
- ρ and σ are two pure states : this means we can write $\rho = |\psi\rangle\langle\psi|$ and $\sigma = |\phi\rangle\langle\phi|$. In this case, we have $F(\rho, \sigma) = |\langle\psi|\phi\rangle|$.

Invariance over unitary operations. The fidelity also has the following property:

Proposition 5.6. *For any two quantum mixed states ρ, σ and any unitary operation U, we have $F(\rho, \sigma) = F(U\rho U^\dagger, U\sigma U^\dagger)$.*

5.2.2 Purifications and Uhlmann's theorem

Our goal here is to introduce the notion of purifications. Then we give an interpretation of fidelity of two states using Uhlmann's theorem.

Purifications

Definition 5.7. *For any state ρ_B, we say that a bipartite state $|\psi_{AB}\rangle$ is a purification of ρ_B is $Tr_A(\psi_{AB}) = \rho$.*

Typically, if two players Alice and Bob share a state $|\psi_{AB}\rangle$ then $\rho_B = Tr_A(|\psi_{AB}\rangle)$ is Bob's reduced density matrix and $|\psi_{AB}\rangle$ is a purification of ρ_B.

For example, if $\rho_B = \frac{3}{4}|0\rangle\langle 0| + \frac{1}{4}|1\rangle\langle 1|$, then $|\psi_{AB}\rangle = \sqrt{\frac{3}{4}}|0\rangle|0\rangle + \sqrt{\frac{1}{4}}|1\rangle|1\rangle$ is a purification of ρ_B. We also have that $|\psi'_{AB}\rangle = \sqrt{\frac{3}{4}}|+\rangle|0\rangle + \sqrt{\frac{1}{4}}|-\rangle|1\rangle$. This means that a state ρ_B can have many purifications.

Fix $\rho_B = \sum_i p_i |e_i\rangle\langle e_i|$. We have that $|\psi_{AB}\rangle = \sum_i \sqrt{p_i}\,|i\rangle\,|e_i\rangle$ is a purification of ρ_B. In fact, for any orthonormal basis $\{|f_i\rangle\}_i$, $|\psi_{AB}\rangle = \sum_i \sqrt{p_i}\,|f_i\rangle\,|e_i\rangle$ is a purification of ρ_B. NB: the above holds even if $\{|e_i\rangle\}_i$ is not a basis.

Uhlmann's Theorems

We now present an interpretation of the fidelity of quantum states

Theorem 5.8 (Uhlmann's first theorem). *For any two states ρ, σ,*

$$F(\rho, \sigma) = \max_{|\psi\rangle, |\phi\rangle} |\,\langle\psi|\phi\rangle\,|$$

where the maximum is taken over purifications $|\psi\rangle$ of ρ and purifications $|\phi\rangle$ of σ.

Theorem 5.9 (Uhlmann's second theorem). *For any two states ρ, σ and any purification $|\psi\rangle$ of ρ, we have*

$$F(\rho, \sigma) = \max_{|\phi\rangle} |\,\langle\psi|\phi\rangle\,|$$

where the maximum is taken over purifications $|\phi\rangle$ of σ.

For example, consider $\rho = \frac{1}{2}|0\rangle\langle 0| + \frac{1}{2}|1\rangle\langle 1|$ and $\sigma = \frac{3}{4}|0\rangle\langle 0| + \frac{1}{4}|1\rangle\langle 1|$. Since ρ and σ are diagonalizable in the same basis, we know that $F(\rho, \sigma) = \sqrt{\frac{3}{8}} + \sqrt{\frac{1}{8}}$. Let $|\psi\rangle = \frac{1}{\sqrt{2}}\,|00\rangle + \frac{1}{\sqrt{2}}\,|11\rangle$ and $|\phi\rangle = \sqrt{\frac{3}{4}}\,|00\rangle + \sqrt{\frac{1}{4}}\,|11\rangle$.

$|\psi\rangle$ (resp. $|\phi\rangle$) is a purification of ρ (resp. σ). Moreover, we have $\langle\psi|\phi\rangle = \sqrt{\frac{3}{8}} + \sqrt{\frac{1}{8}}$. These purifications are optimal with regards to Uhlmann's theorem.

5.2.3 Angle distance

As we said previously, the quantity $1 - F$ is not a distance since it doesn't satisfy the triangle inequality. Our goal here is to construct a distance out of the fidelity.

Definition 5.10. *For any two quantum states ρ, σ, we define their angle as $Angle(\rho, \sigma) = Arccos(F(\rho, \sigma))$*

Fix two pure states $|\psi\rangle$ and $|\phi\rangle$ with $|\,\langle\psi|\phi\rangle\,| = \cos(\alpha)$, then $Angle(|\psi\rangle\langle\psi|, |\phi\rangle\langle\phi|) = \alpha$. The notion of angle for mixed states somehow extends the notion of angle that exists for pure states.

The angle is a distance Indeed, it satisfies the following properties

- $Angle(\rho, \sigma) = 0 \Leftrightarrow \rho = \sigma$
- $0 \leq Angle(\rho, \sigma) \leq \pi/2$
- $Angle(\rho, \sigma) = Angle(\sigma, \rho)$
- $Angle(\rho, \tau) \leq Angle(\rho, \sigma) + Angle(\sigma, \tau)$

5.2.4 Fuchs - Van de Graaf inequalities

Finally, we present a relationship between the trace distance of two quantum states and the fidelity of those states.

Proposition 5.11 ([FG99])**.** *For any states ρ, σ, we have*

$$(1 - F(\rho, \sigma) \leq \Delta(\rho, \sigma) \leq \sqrt{1 - F^2(\rho, \sigma)}$$

or conversely

$$(1 - \Delta(\rho, \sigma) \leq F(\rho, \sigma) \leq \sqrt{1 - \Delta^2(\rho, \sigma)}$$

Chapter 6

First quantum cryptographic protocols

6.1 Bit commitment

A bit commitment scheme is a protocol between two parties Alice and Bob, denoted hereafter A and B. A bit commitment scheme consists of 2 phases; a *commit phase* and a *reveal phase*.

- At the commit phase, Alice commits to a bit $b \in \{0, 1\}$ and Bob should not be able to guess b at the end of the commit phase.
- At the reveal phase, Alice reveals b. She shouldn't be able to change her mind about the bit b she reveals.

Security requirements:

- *Completeness:* If both players are honest, the protocol should succeed wp. 1.
- *Hiding property:* If Alice is honest and Bob is dishonest, his cheating probability is

  $$P_B^* = \Pr[\text{ Bob guesses } b \text{ after the commit phase }].$$

- *Binding property:* If Alice is dishonest and Bob is honest, her cheating probability is

  $$P_A^* = \frac{1}{2}\left(\Pr[\text{ Alice successfully reveals } b = 0] + \Pr[\text{ Alice successfully reveals } b = 1]\right).$$

  for the same commit phase. This means that after the commit phase, we want to bound Alice possibility to reveal both $b = 0$ and $b = 1$ successfully.

6.1.1 Generic example of commitment schemes

Let $|\psi_{AB}^0\rangle$ and $|\psi_{AB}^1\rangle$ two quantum bipartite states. Consider the following protocol

- **Commit phase:** Alice wants to commit to a bit b. She creates $\left|\psi^b_{AB}\right\rangle$ and sends the B part to Bob. After the commit phase, Bob has $\rho_b = Tr_A(\left|\psi^b_{AB}\right\rangle)$.
- **Reveal phase:** Alice sends the A part of the quantum state $\left|\psi^b_{AB}\right\rangle$ as well as b. Bob checks that he has $\left|\psi^b_{AB}\right\rangle$ by projecting the state he has onto $\left|\psi^b_{AB}\right\rangle$.

Cheating probabilities We define the cheating probabilties for the two players:

- $P^*_A = \max(\Pr[\text{Alice cheats}]) = \max\left(\frac{1}{2}\Pr[\text{Alice reveals } b=0] + \frac{1}{2}\Pr[\text{Alice reveals } b=1]\right)$ (for the same commit phase)
- $P^*_B = \max(\Pr[\text{Bob cheats}]) = \max(\Pr[\text{Bob can guess b after the commit phase}])$

6.1.2 Cheating strategies

Cheating Bob : He has ρ_b after the commit phas and tries to guess b. We have that

$$P^*_B = \Pr[\text{Bob can guess } b] = \frac{1}{2} + \frac{\Delta(\rho_0,\rho_1)}{2}$$

Cheating Alice: Fix a cheating strategy for Alice and let σ the state that Bob has after the commit phase. During the reveal phase, is she reveals $b = 0$ then she sends qubits such that Bob has a pure state $|\phi_0\rangle$. If she reveals $b = 1$, then she sends qubits such that Bob has a pure state $|\phi_1\rangle$.

We have $Tr_A(|\phi_0\rangle) = Tr_A(|\phi_1\rangle) = \sigma$. If Alice reveals $b = 0$, we have

$$\Pr[\text{Bob accepts } |b=0] = |\left\langle\phi_0|\psi^b_{AB}\right\rangle|^2$$

If Alice reveals $b = 1$, we have

$$\Pr[\text{Bob accepts } |b=1] = |\left\langle\phi_1|\psi^b_{AB}\right\rangle|^2$$

Using Uhlmann's theorem, we have

$$\max_{|\phi_0\rangle} |\left\langle\phi_0|\psi^0_{AB}\right\rangle|^2 = F^2(\sigma,\rho_0)$$

where the maximum is taken over purifications $|\phi_0\rangle$ of σ. We also have

$$\max_{|\phi_1\rangle} |\left\langle\phi_1|\psi^1_{AB}\right\rangle|^2 = F^2(\sigma,\rho_1)$$

where the maximum is taken over purifications $|\phi_1\rangle$ of σ.

This gives us

$$\frac{1}{2}\left(\Pr[\text{Bob accepts } |b=0] + \Pr[\text{Bob accepts } |b=1]\right) = \frac{1}{2}F^2(\sigma,\rho_0) + \frac{1}{2}F^2(\sigma,\rho_1)$$

Since Alice can choose any σ, we have

$$P^*_A = \max_\sigma\left(\frac{1}{2}F^2(\sigma,\rho_0) + \frac{1}{2}F^2(\sigma,\rho_1)\right)$$

Recall also that

$$P_B^* = \frac{1}{2} + \Delta(\rho_0, \rho_1)/2$$

We want to remove the maximization for Alice's cheating probability. We use the following Lemma

Lemma 2.

$$\forall \sigma, \frac{1}{2}F^2(\sigma, \rho_0) + \frac{1}{2}F^2(\sigma, \rho_1) \leq \frac{1}{2}\left(1 + F(\rho_0, \rho_1)\right)$$

Proof. Use the Angle distance (proof skipped here) □

Also, there exists a σ such that $\frac{1}{2}F^2(\sigma, \rho_0) + \frac{1}{2}F^2(\sigma, \rho_1) = \frac{1}{2}\left(1 + F(\rho_0, \rho_1)\right)$. From there, we conclude that

$$P_A^* = \frac{1}{2} + \frac{F(\rho_0, \rho_1)}{2}$$
$$P_B^* = \frac{1}{2} + \frac{\Delta(\rho_0, \rho_1)}{2}$$

Best coin flipping protocols of this type By using the Fuchs - Van de Graaf inequalities, we have $F(\rho_0, \rho_1) \geq 1 - \Delta(\rho_0, \rho_1)$. This implies $P_A^* + P_B^* \geq 3/2$ or $\max\{P_A^*, P_B^*\} \geq 3/4$. Is this tight ? Yes

Consider the states $\rho_0 = \frac{1}{2}|0\rangle\langle 0| + \frac{1}{2}|2\rangle\langle 2|$ and $\rho_1 = \frac{1}{2}|1\rangle\langle 1| + \frac{1}{2}|2\rangle\langle 2|$. We can calculate

$$\Delta(\rho_0, \rho_1) = \frac{1}{2}(|1/2 - 0| + |0 - 1/2| + |1/2 - 1/2|) = 1/2$$
$$F(\rho_0, \rho_1) = \sqrt{1/2 \cdot 0} + \sqrt{0 \cdot 1/2} + \sqrt{1/2 \cdot 1/2} = 1/2$$

NB: This analysis covers only quantum bit commitment protocols for specific commit/reveal phases. This is not the most general analysis. In fact, there exists interactive quantum BC protocols with cheating probabiltiies $< 3/4$.

6.2 Bit commtiment based coin flipping

Here, we show that any bit commitment protocols with cheating probabilities P_A^*, P_B^* can be transformed into a quantum bit commitment scheme with the same cheating probabilities.

Protocol for QCF using QBC

1. Alice picks a random $a \in \{0, 1\}$. Then, she commits to a using the QBC protocol.
2. Bob sends a random $b \in \{0, 1\}$ and sends b to Alice.
3. Alice reveals a, as described in the QBC protocol.
4. The output of the coin is $c = a \oplus b$.

We can see that

Pr[Bob cheats in the CF protocol] = Pr[Bob guesses a after step 1] = Pr[Bob cheats in the BC protocol].

and

$$\begin{aligned}\text{Pr[Alice cheats in the CF protocol]} &= \frac{1}{2}\,\text{Pr[Alice cheats in the CF protocol |Bob sends } b=0] + \\ &\quad \frac{1}{2}\,\text{Pr[Alice cheats in the CF protocol |Bob sends } b=1] \\ &= \frac{1}{2}\,\text{Pr[Alice successfully reveals } a=0] + \frac{1}{2}\,\text{Pr[Alice successfully reveals } a=1] \\ &= \text{Pr[Alice can cheat in the BC protocol]}\end{aligned}$$

NB: On the other hand, we don't have $QCF \Rightarrow QBC$.

6.3 Quantum Random Access codes

A quantum encoding $x \in \{0,1\}^n \to |\psi_x\rangle$ on m qubits is called a $(n,m,p)-QRAC$, if one can recover any bit x_i with probability p when having access to $|\psi_x\rangle$. $(n,m,1)-QRACs$ are impossible for $m < n$.

Construction of a $(2,1,\cos^2(\pi/8))$**-QRAC** We consider the encoding $|\psi_{00}\rangle = |0\rangle$, $|\psi_{01}\rangle = |+\rangle$, $|\psi_{10}\rangle = |-\rangle$, $|\psi_{11}\rangle = |1\rangle$.

- If I want to learn x_1, I measure in the $\{|v\rangle, |v^\perp\rangle\}$ basis with $|v\rangle = \cos(\pi/8)|0\rangle + \sin(\pi/8)|1\rangle$ and $|v^\perp\rangle = \sin(\pi/8)|0\rangle - \cos(\pi/8)|1\rangle$.
- If I want to learn x_2, I measure in the $\{|w\rangle, |w^\perp\rangle\}$ basis with $|w\rangle = \cos(\pi/8)|0\rangle - \sin(\pi/8)|1\rangle$ and $|w^\perp\rangle = \sin(\pi/8)|0\rangle + \cos(\pi/8)|1\rangle$.

We have

$$\begin{gathered}|\langle v|\psi_{00}|\rangle^2 = |\langle v|\psi_{01}\rangle|^2 = \cos^2(\pi/8) \\ |\langle v^\perp|\psi_{10}\rangle|^2 = |\langle v^\perp|\psi_{11}\rangle|^2 = \cos^2(\pi/8) \\ |\langle w|\psi_{00}\rangle|^2 = |\langle v|\psi_{10}\rangle|^2 = \cos^2(\pi/8) \\ |\langle w^\perp|\psi_{01}\rangle|^2 = |\langle w^\perp|\psi_{11}\rangle|^2 = \cos^2(\pi/8)\end{gathered}$$

which shows that this construction is indeed a $(2,1,\cos^2(\pi/8))$-QRAC.

NB: These measurements are optimal.

Chapter 7

Quantum key distribution

Key distribution is an important cryptographic primitive, which is defined as follows.

Key distribution

- Alice and Bob communicate over a *public* and *authenticated* channel.
- At the end of the scheme, they should agree on a key $K \in \{0,1\}^k$.
- Any adversary eavesdropping and tampering the channel shouldn't be able to have any (or vanishingly little) information about K.

7.1 Encoding bits of key inside qubits

Alice has a string $K = k_1, \ldots, k_n$ which we call the initial key. Her goal is to transmit the bits of K to Bob in a way that can't be intercepted without being caught. For each i, She performs the following encoding:

The BB84 encoding of a bit k_i

- Pick a random $b_i \in \{0,1\}$.
- If $b_i = 0$, construct $|\psi_i\rangle = |k_i\rangle$. If $b_i = 1$, construct $|\psi_i\rangle = H\,|k_i\rangle$.
- Output $|\psi_i\rangle$.

This encoding is very simple. You pick a random $b_i \in \{0,1\}$, and you encode k_i in the computational basis if $b_i = 0$ and in the Hadamard basis if $b_i = 1$.

k_i	b_i	$\lvert\psi_i\rangle$
0	0	$\lvert 0\rangle$
0	1	$\lvert +\rangle$
1	0	$\lvert 1\rangle$
1	1	$\lvert -\rangle$

The full protocol is then the following

The BB84 protocol

- Alice picks a random initial raw key $K = k_1, \ldots, k_n$ uniformly at random.
- For each $i \in \{1, \ldots, n\}$, Alice picks a random $b_i \in \{+, \times\}$, constructs $|\psi_i\rangle = |k_i\rangle^{b_i}$ and sends $|\psi_i\rangle$ to Bob.
- Bob picks some random basis $b'_1, \ldots, b'_n \in \{+, \times\}$ and measures each qubit $|\psi_i\rangle$ in the b'_i basis. Let c_i be the outcome of this measurement.
- Bob sends to Alice the basis $\mathbf{b}' = b'_1, \ldots, b'_n$ he used for his measurements using a public channel. Alice sends back the subset $I = \{i \in [n] : b_i = b'_i\}$ to Bob.
- Alice then picks a random subset $J \subseteq I$ of size $\frac{|I|}{2}$ which is the subset of indices for which Alice and Bob check that there wasn't any interception and sends J to Bob. For $j \in J$, Alice also sends k_j to Bob.
- For each $j \in J$, Bob checks that $k_j = c_j$. If one of these checks fail, he aborts.
- Let $L = I \backslash J = l_1, \ldots, l_{|L|}$ be the subset of indices used for the final raw key. We write $K_A = \{k_l\}_{l \in L}$ and $K_B = \{c_l\}_{l \in L}$.
- Alice and Bob perform key reconciliation to agree on a key K_{raw}.
- They perform privacy amplification to ensure that Alice has no information about the key.

7.1.1 Key reconciliation

The idea of key reconciliation is that $K_A \in \{0,1\}^m$ is usually different from K_B. How does that happen? There are two possible scenarios:

- An eavesdropper only intercepted a small number of qubits (so he wasn't caught with some constant probability), but disturbed the signal enough st. there is i st. $k_i \neq c_i$ for $i \in I \backslash J$.
- Hardware imperfection in the signal transmission and in the measurement create some inconsistency.

In order to perform key reconciliation the idea is to use a binary error-correcting code. For our purposes, an error correcting code is a set $\mathcal{C} \subseteq \{0,1\}^m$ st. $\min_{x,y \neq x \in \mathcal{C}} |x - y|_H = d$ for a parameter d of the code called the minimal distance. Alice chooses a code $\mathcal{C}$ st. $K_A \in \mathcal{C}$. This means that if $|K_B - K_A| \leq \frac{d}{2}$ then Bob can recover K_A from K_B since it is the unique element of $\mathcal{C}$ at distance at most $\frac{d}{2}$. Here are the challenges of this method.

- We must choose a code $\mathcal{C}$ with a large enough minimal distance d such that $|K_B - K_A| < \frac{d}{2}$.
- However, the adversary now knows that $K_A \in \mathcal{C}$ so the size of $\mathcal{C}$ must remain very large. There is a trade-off between the size of $\mathcal{C}$ and the minimal distance d.
- Even if the decoding is unique, it has to be computationally efficient. Even if it is unique, recovering K_A from K_B can be a very difficult task. For example, if we take a random code $\mathcal{C}$, this task is NP-hard.

Basic BB84 protocol

Alice		**Bob**
$k_1, \ldots, k_n \leftarrow\!\$ \ \{0,1\}$		
$b_1, \ldots, b_n \leftarrow\!\$ \ \{+,\times\}$		
	$\xrightarrow{\lvert k_1\rangle^{b_1}, \ldots, \lvert k_n\rangle^{b_n}}$	
		$b'_1, \ldots, b'_n \leftarrow\!\$ \ \{+,\times\}$
		$\forall i,\ c_i \leftarrow$ measure $\lvert k_i\rangle^{b_i}$ in basis b'_i
	$\xleftarrow{b'_1, \ldots, b'_n}$	
$I = \{i : b_i = b'_i\}$		
$J \leftarrow\!\$ \ \{S \subseteq I : \lvert S\rvert = \frac{\lvert I\rvert}{2}\}$		
	$\xrightarrow{I, J, \{k_j\}_{j \in J}}$	
		Check that $\forall j \in J, k_j = c_j$
	$\xleftarrow{\text{“Check passed”}}$	
$K_A = \{k_l\}_{l \in I \setminus J}$		$K_B = \{c_l\}_{l \in I \setminus J}$
	$\xrightarrow{\text{Key reconciliation}}$	
	$\xleftarrow{\qquad\qquad}$	
Agree on $K \in \{0,1\}^{k'}$		Agree on $K \in \{0,1\}^{k'}$
	$\xrightarrow{\text{Agree on } h}$	
	$\xleftarrow{\qquad\qquad}$	
$K_{\text{final}} = h(K)$		$K_{\text{final}} = h(K)$

Figure 7.1: Description of a basic BB84 quantum key distribution protocol

There are varieties of choices for this task, for example using LDPC codes.

7.1.2 Privacy amplification

At the end of the reconciliation phase, the eavesdropper Eve could still have a little bit of information about K. In order to construct K_{final}, we apply a hash function to ensure that this information is destroyed.

Chapter 8

Quantum information theory

8.1 Classical entropy

Entropy is arguably one of the most important concepts in information theory.

Definition 8.1. *Let $p = (p_1, \dots, p_n)$ be a discrete probability function, so $p_i \geq 0$ and $\sum_i p_i = 1$. The entropy $H(p)$ of p is defined as*

$$H(p) = \sum_{i=1}^{n} -p_i \log_2(p_i).$$

The entropy $H(p)$ measures the amount of uncertainty in p. For example, for $p = (1, 0, \dots, 0)$, we have $H(p) = 0$. For $p = (\frac{1}{2}, \frac{1}{2}, 0, \dots, 0)$ then $H(p) = 1$. If $p = (\frac{1}{n}, \dots, \frac{1}{n})$ then $H(p) = \log_2(n)$ (maximal). The entropy can be informally seen as the amount of coins required to mimic p.

Other example: $p(0) = \frac{3}{4}, p(1) = \frac{1}{4}$ so $H(p) = \frac{3}{4}\log(4/3) + \frac{1}{4}\log(1/4) \approx 0.811$. It doesn't seem we can send i with strictly less than 1 bit. How do we interpret $H(p) < 1$ as a noiseless compression bound? Consider $p^2(xy) = p(x)p(y)$ for $x, y \in \{0, 1\}$. We have $H(p^2) = 2H(p)$. Now, if Alice has i, j wp. $p^2(i, j)$ and wants to send these 2 bits, she can do the following:

1. If $(x, y) = (0, 0)$: send "0".
2. If $(x, y) = (0, 1)$: send "01".
3. If $(x, y) = (1, 0)$: send "011".
4. If $(x, y) = (1, 1)$: send "111".

We have

$$\text{Average amount of bits sent } = \frac{9}{16} + 2 * \frac{3}{16} + 3 * (\frac{3}{16} + \frac{1}{16}) = \frac{27}{16} = 1.6875.$$

and we have $H(p^2) \leq \frac{27}{16} < 2$. If we take p^n, we can find an encoding st. the average number of bits sent will be closer and closer to $H(p)$. This is Shannon's noiseless source coding theorem.

8.1.1 Quantum entropy

Quantum Shannon entropy: $S(\cdot)$.

Definition 8.2. *Let $\rho = \sum_i \lambda_i |e_i\rangle\langle e_i|$ be a quantum mixed state with it's* ***spectral*** *decomposition (so the $|e_i\rangle$ are pairwise orthogonal and have norm 1). We define the Shannon entropy of ρ as*

$$S(\rho) := H((\lambda_1, \dots, \lambda_n)) = \sum_i -\lambda_i \log(\lambda_i).$$

We can also write $S(\rho) = -Tr(\rho \log(\rho))$ where $\log(\rho) = \sum_i \log(\lambda_i) |e_i\rangle\langle e_i|$. $\log(\rho)$ is a hermitian matrix but not a quantum state!

- If ρ is in some register R, we will sometimes equivalently write $S(R)_\rho$ instead of $S(\rho)$.
- If we take ρ_{AB} in registers AB and $\rho_A = Tr_B(\rho_{AB})$, we will equivalently write

$$S(\rho_A) = S(A)_{\rho_A} = S(A)_{\rho_{AB}}.$$

8.1.2 Properties of the quantum entropy

- Let $\rho = \sum_i \lambda_i |e_i\rangle\langle e_i|$ a quantum mixed state with it's spectral decomposition.
- Let U a quantum unitary and let $|f_i\rangle = U(|e_i\rangle)$. Recall that applying U to ρ gives the state $U\rho U^\dagger = \sum_i \lambda_i |f_i\rangle\langle f_i|$.
- We immediately have $S(U\rho U^\dagger) = S(\rho)$.
- $S(\rho) \geq 0$.
- $S(A)_{\rho_{AB}} - S(B)_{\rho_{AB}} \leq S(AB)_{\rho_{AB}} \leq S(A)_{\rho_{AB}} + S(B)_{\rho_{AB}}$.
- $S(\rho_A \otimes \rho_B) = S(\rho_A) + S(\rho_B)$.

Proposition 8.3. *Let ρ be a quantum state. Let $\Pi : \{\Pi_1, \dots, \Pi_n\}$ be a projective measurement and let $p_i = tr(\rho \Pi_i)$. We have*

$$S(\rho) \leq H(p).$$

$$S(\sum_i \Pi_i \rho \Pi_i) = tr(\sum_i \Pi_i \rho \Pi_i \log(\sum_i \Pi_i \rho \Pi_i)) \tag{8.1}$$

8.1.3 Conditional quantum entropy and conditional mutual information

Definition 8.4. *The conditional entropy $S(A|B)$ is defined as*

$$S(A|B)_{\rho_{AB}} := S(AB)_{\rho_{AB}} - S(B)_{\rho_{AB}}.$$

- Unlike classical conditional entropy, the quantum one can be negative! Take for example $\rho_{AB} = |\Phi^+\rangle\langle\Phi^+|$. We have $S(AB)_{\rho_{AB}} = 0$ and $S(A)_{\rho_{AB}} = S(B)_{\rho_{AB}} = 1$.

- Chain rule:
$$S(A|C)_{\rho_{ABC}} + S(B|AC)_{\rho_{ABC}} = S(AB|C)_{\rho_{ABC}}.$$
- We also have $S(A|B)_{\rho_{ABC}} \leq S(A)_{\rho_{ABC}}$ and $S(A|BC)_{\rho_{ABC}} \leq S(A|B)_{\rho_{ABC}}$. This implies
$$S(A|C)_{\rho_{ABC}} + S(B|C)_{\rho_{ABC}} \geq S(A|C)_{\rho_{ABC}} + S(B|AC)_{\rho_{ABC}} = S(AB|C)_{\rho_{ABC}}. \tag{8.2}$$

Definition 8.5. *The mutual information $I(A : B)$ is defined as*

$$I(A : B)_{\rho_{AB}} = S(A)_{\rho_{AB}} + S(B)_{\rho_{AB}} - S(AB)_{\rho_{AB}}.$$

Definition 8.6. *The conditional mutual information $I(A : B|C)$ is defined as*

$$I(A : B|C)_{\rho_{ABC}} = S(A|C)_{\rho_{ABC}} + S(B|C)_{\rho_{ABC}} - S(AB|C)_{\rho_{ABC}}.$$

We have that $I(A : B)_{\rho_{AB}} \geq 0$ and $I(A : B|C)_{\rho_{ABC}} \geq 0$ from Equation 8.2.

Proposition 8.7 (Pinsker's inequality)**.** *For a quantum state ρ_{AB}, we have*

$$I(A : B) \geq \frac{1}{2}\Delta^2(\rho_{AB}, \rho_A \otimes \rho_B).$$

8.1.4 The index problem

The index problem; Alice has a uniformly random string $x \in \{0,1\}^n$. Bob is given a uniformly random index i. Alice and Bob cooperate and the goal of the index game is for Bob to output x_i. Without any communication, Bob can't do better than randomly guess x_i so he will succeed wp. $\frac{1}{2}$.

Now, assume Alice sends a quantum pure state $|\psi_x\rangle$ of $m < n$ qubits to Bob. An m-qubit state consists of 2^m complex numbers so it shouldn't be hard to encode the information of each x in a different state $|\psi_x\rangle$. However, when Bob receives $|\psi_x\rangle$, he can't recover all the amplitudes of $|\psi_x\rangle$ and he is limited by the laws of quantum measurements in order to recover x.

We can actually show that for $m < n$, Bob cannot recover perfectly x_i. We will also give quantitative versions of this result.

Bounding the probability of winning $Index_n$ with m bits of communication Let p_i the probability of winning when Bob has input i. After Alice sends her message, Alice and Bob share the state

$$\rho = \frac{1}{2^n} \sum_{x \in \{0,1\}^n} |x\rangle\langle x|_A |\psi_x\rangle\langle\psi_x|_B.$$

We have

$$I(A : B)_\rho = S(A)_\rho + S(B)_\rho - S(AB)_\rho = n + S(B)_\rho - n = S(B)_\rho \leq m.$$

We write $A = X_1, \ldots, X_n$, and

$$\begin{aligned} I(A:B) = S(A) - S(A|B) = S(A) - S(X_1,\ldots,X_n|B) \geq n - \sum_{i=1}^{n} S(X_i|B) \\ = \sum_{i=1}^{n} 1 - S(X_i|B) = \sum_{i=1}^{n} S(X_i) - S(X_i|B) = \sum_{i=1}^{n} I(X_i:B) \end{aligned}$$

which gives

$$\frac{1}{n}\sum_{i=1}^{n} I(X_i:B) \leq \frac{m}{n}.$$

Bob can perform a measurement on his part that guesses x_i wp. p_i on input i. So after his guess, the state is

$$\rho^2 = \frac{1}{n2^n} \sum_{\substack{x\in\{0,1\}^n \\ i\in[n]}} |x\rangle\langle x|_A \otimes \Big(|i\rangle\langle i|_I \otimes \Big(p_i|x_i\rangle\langle x_i|_G \otimes \zeta_E^{x,i} + (1-p_i)|\overline{x_i}\rangle\langle\overline{x_i}|_G \otimes \widetilde{\zeta}_E^{x,i}\Big)\Big).$$

where there registers I, G, E are on Bob's side. We have

$$I(X_i:B)_{\rho^2} \geq I(X_i:G)_{\rho^2} = 1 + 1 - (1 + H_2(p)) = 1 - H_2(p_i)$$

with $H_2(p) = -p\log(p) - (1-p)\log(1-p)$. Here, we use

- $\rho^2_{X_i} = \frac{1}{2}\left(|0\rangle\langle 0| + |1\rangle\langle 1|\right).$
- $\rho^2_G = \frac{1}{2}\left(|0\rangle\langle 0| + |1\rangle\langle 1|\right).$
- $\rho^2_{X_iG} = \frac{1-p_i}{2}\left(|00\rangle\langle 00| + |11\rangle\langle 11|\right) + \frac{p_i}{2}\left(|01\rangle\langle 01| + |10\rangle\langle 10|\right)$. So $H(X_iG)_{\rho^2} = 1 + H_2(p_i)$.

Putting everything together, we have

$$m \geq \sum_{i=1}^{n}(1 - H(p_i)).$$

This gives interesting information. For example, if we want Bob to win wp. 1, we have necessarily $m \geq n$ meaning that we cannot perform better than sending the whole string x. Moreover, if the players want that Bob always succeeds wp. p for each i, we have necessarily $m \geq n - nH(p)$. These bounds are not necessarily tight.

Bibliography

[FG99] Christopher A. Fuchs and Jeroen Van De Graaf. Cryptographic distinguishability measures for quantum-mechanical states. *IEEE Trans. Inform. Theory 45. No*, pages 45–1216, 1999.

[NC00] Michael A. Nielsen and Isaac L. Chuang. *Quantum computation and quantum information.* Cambridge University Press, New York, NY, USA, 2000.