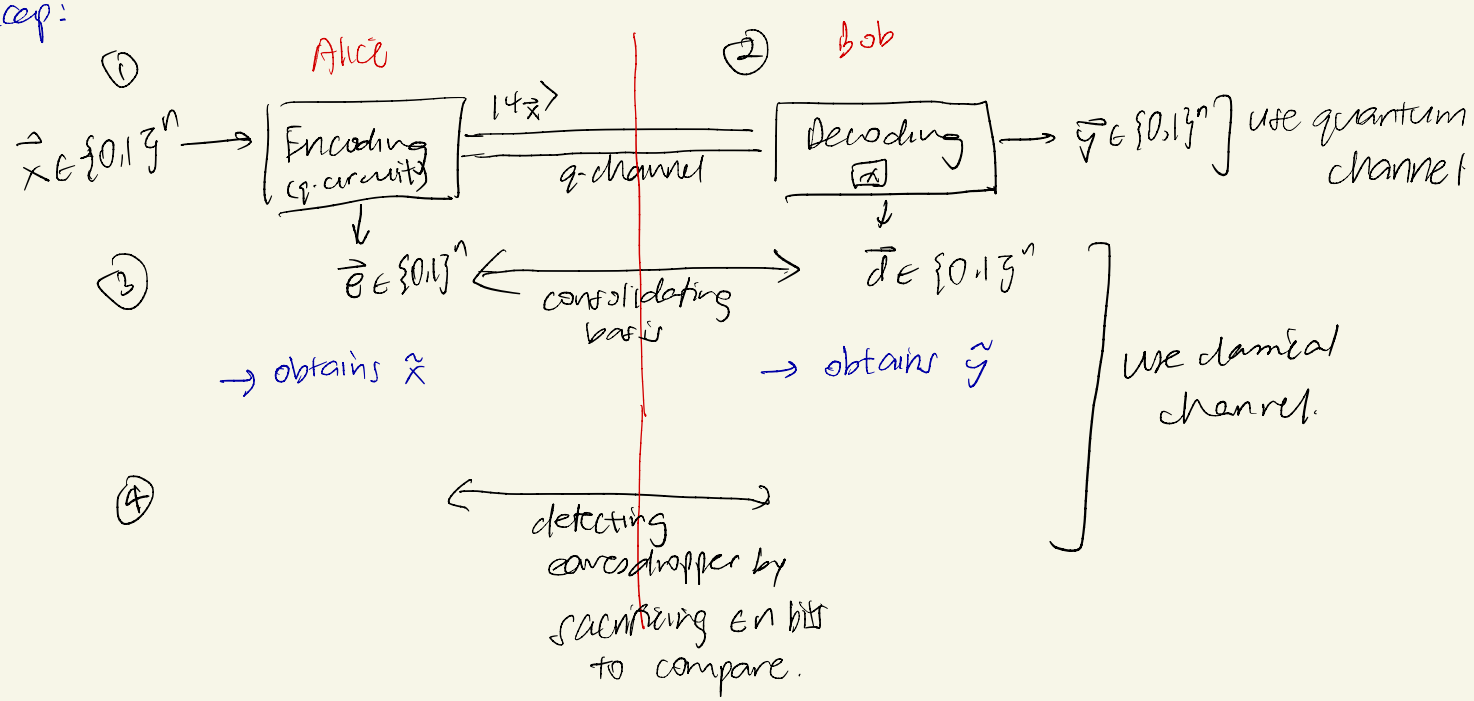
Dealing with noise in QKD via error correction

Recap:



Problem: when there is noise in the quantum channel, even with no eavesdropper, this will flip the bits between $\tilde{x}$ and $\tilde{y}$ anyway, which may cause A & B to falsely abort the protocol in step 4.

Possible solutions:

- At lower noise rates:

If the expected # bits flipped by the noise is $\delta_1 n$ and the expected # bits flipped by the eavesdropper is $\delta_2 n$ and

$$\delta_2 >> \delta_1$$

we can simply accept in step 4 if $\geq (1 - \frac{3}{2}\delta_1)n$ bits agree.

— At higher noise rates: one way is to use error correction.

Error correction primer

A binary linear code is a subspace $\mathcal{C} \subseteq \mathbb{F}_2^m$, with $|\mathcal{C}| = 2^k$.

Its elements are called codewords.

Def: Distance of an error-correcting code (ECC)

= minimum number of bits differing between any two codewords.

Example: repetition code of length 3.

To communicate a message of 1 bit, $b \in \{0,1\}$, I encode it as a 3-bit codeword, bbb.

$\mathcal{C} = \{000, 111\}$, $k = 1$, $d = 3$ (∵ 000, 111 differ at 3 positions)

ECCs have a nice property: a corrupted codeword can be restored to its correct version, via a process known as decoding.

Def Decoding:

Input: corrupted codeword

$y = c \oplus e$ where $c \in \mathcal{C}$

↑ bitwise addition mod 2

e = some error vector, 1 at the bits that were flipped and 0 otherwise.

Output: c (correct codeword)

Many different decoding algorithms exist, depending on the code!

Example: to decode the repetition code, I can simply take a majority vote amongst the bits of the corrupted codeword.

E.g. $b\bar{b}b \xrightarrow[\text{vote}]{\text{majority}} b$

(2 votes for b, 1 vote for $\bar{b}$)

Of course, sometimes if the corruption flips too many bits, this process fails.

Useful fact about decoding: If $y = c \oplus e$ and $\text{weight}(e) \leq t$ (weight = number of 1's) where $t = \lfloor \frac{d-1}{2} \rfloor$, where d = distance of $\mathcal{C}$, then it is always possible to decode, e.g. by exhaustive search, that is:

Find $x \in \mathcal{C}$ s.t. $\text{weight}(x \oplus y)$ is minimal.

Exercise: prove this.

Continuing. At higher noise rates we can use:

QKD protocol with error correction

(not a formal proof)

Setting:

- After step (3), Alice holds $\tilde{x}$ and Bob holds $\tilde{y}$.
- Suppose we expect ϵ fraction of bits between $\tilde{x}$ and $\tilde{y}$ to differ due to noise of quantum channel, in the absence of eavesdropper, ie we expect

$$\tilde{x} - \tilde{y} = \tilde{x} \oplus \tilde{y} =: N \quad \text{where} \quad \text{weight}(N) \approx \epsilon n$$

addition is the same as subtraction, mod 2

(We can think of N as an "error vector", indicating where the noise flipped the bits.)

New protocol: Insert the following step (3'), after step (3) and before step (4).

- Alice chooses an error-correcting code $\mathcal{C}$, ~~that~~ will correct correct ϵ' fraction of the bits, $\epsilon' > \epsilon$, and announces this over the channel.

- she chooses any codeword, $c \in \mathcal{C}$, and sends

$$\tilde{x} \oplus c$$

to Bob, over their shared public channel.

- Bob subtracts $\tilde{y}$ from this, getting

$$\tilde{x} \oplus c - \tilde{y} = (\tilde{x} \oplus \tilde{y}) \oplus c = N \oplus c$$

- Since N is 1 only in the locations where noise acted, we can think of it as an error vector. This puts us in the decoding setting (recall the definition!)

Bob applies decoding to $N \oplus c$, obtaining c, and N. This tells him exactly where the noise acted.

- Bob computes

$$\tilde{y} \oplus N = \tilde{y} \oplus \tilde{x} \oplus \tilde{y} = \tilde{x}$$ (same message as Alice)