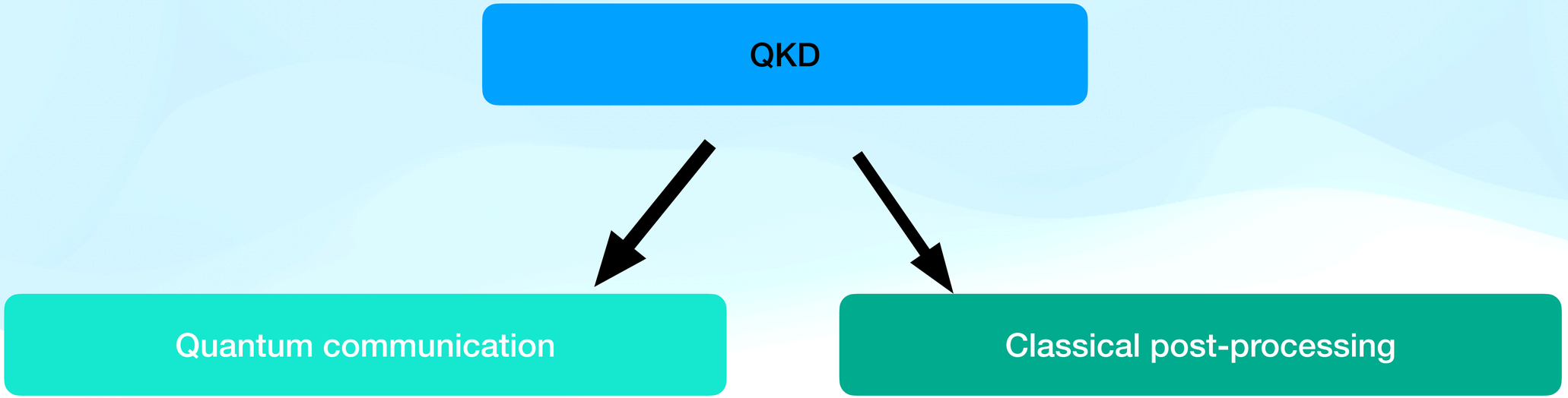
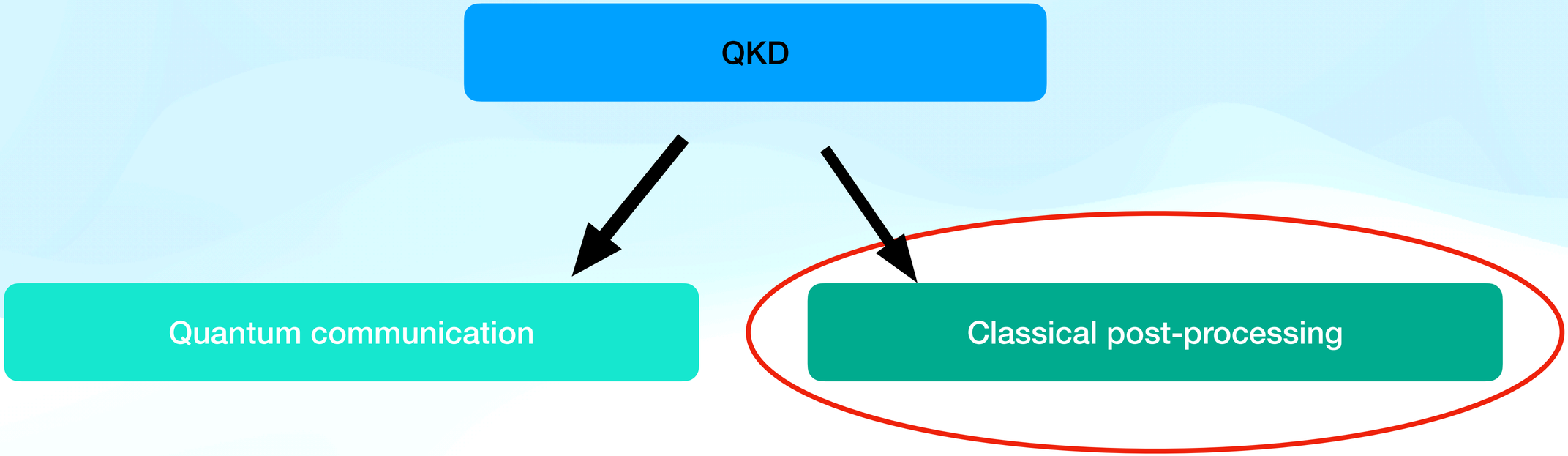
Classical post processing

Quantum key distribution



Quantum key distribution



Classical post-processing

Step 1: **Parameter estimation** - estimate error rate

- bound Eve’s information gain on the string

Classical post-processing

Step 1: **Parameter estimation** - estimate error rate

- bound Eve's information gain on the string

Step 2: Error correction - Make shared strings identical

Classical post-processing

Step 1: **Parameter estimation** - estimate error rate

- bound Eve's information gain on the string

Step 2: **Error correction** - Make shared strings identical

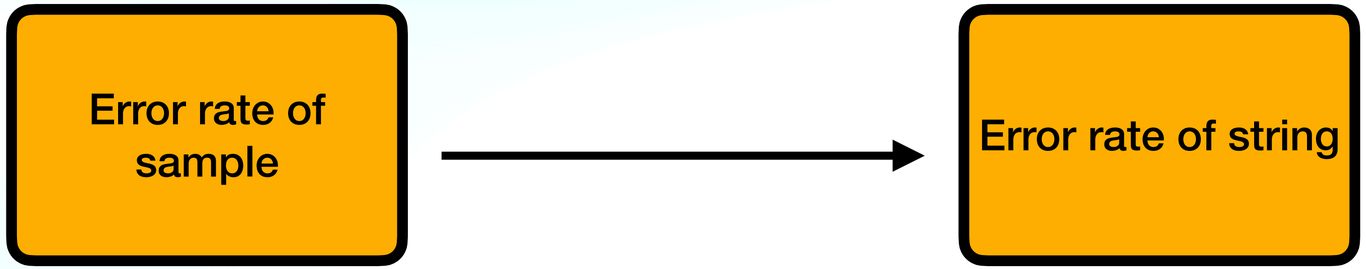
Step 3: **Privacy amplification** - minimise Eve's information on shared corrected strings.

Parameter estimation

Goal: Estimate the error rate in the bit string.

Standard procedure:

1. Alice sends a small sample of her string to Bob.
2. Bob compares it to his string and estimates the error rate.

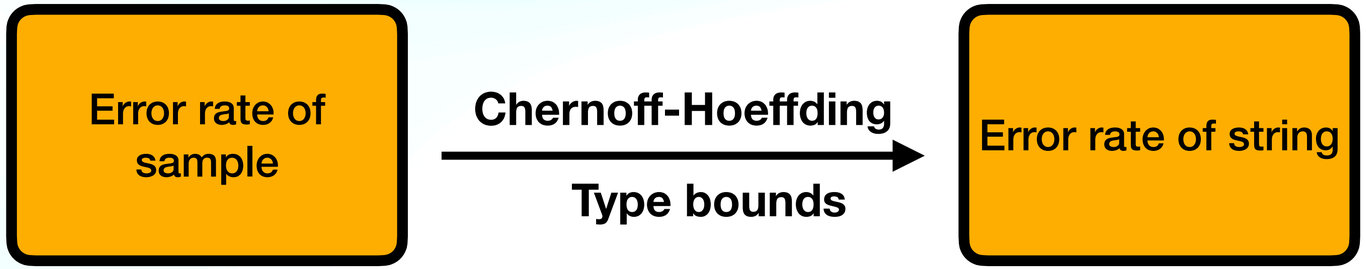


Parameter estimation

Goal: Estimate the error rate in the bit string.

Standard procedure:

1. Alice sends a small sample of her string to Bob.
2. Bob compares it to his string and estimates the error rate.

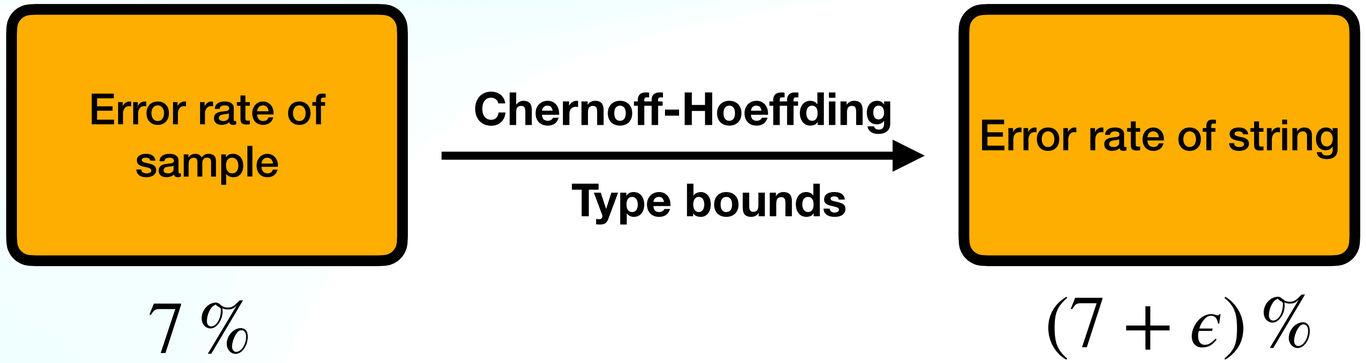


Parameter estimation

Goal: Estimate the error rate in the bit string.

Standard procedure:

1. Alice sends a small sample of her string to Bob.
2. Bob compares it to his string and estimates the error rate.

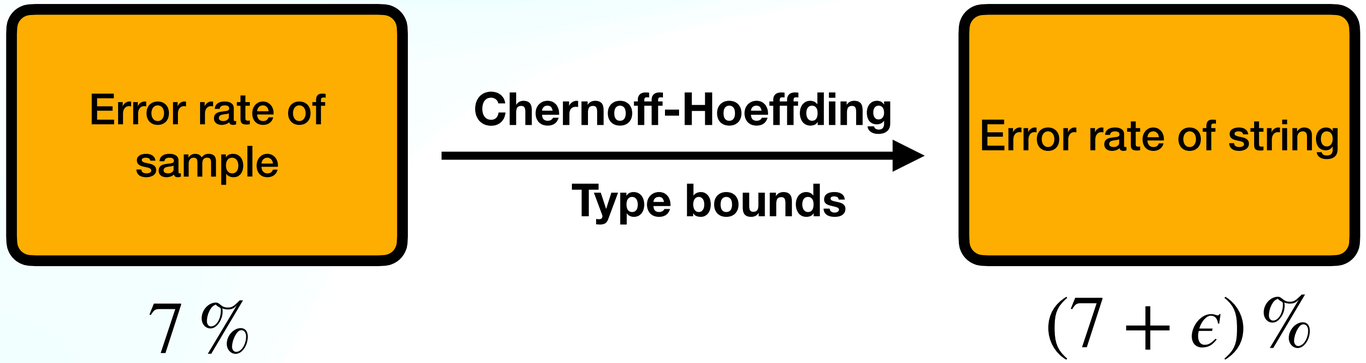


Parameter estimation

Goal: **Estimate the error rate in the bit string.**

Standard procedure:

1. **Alice sends a small sample of her string to Bob.**
2. **Bob compares it to his string and estimates the error rate.**



Basic question: How well does the error rate of a sample represent the error rate of the entire string?

Serfling's inequality

For a set of N random variables K_i with values $k_i \in \{0,1\}$, where $i \in \{1,\cdots,N\}$, the average is defined as

$$K = \frac{1}{N}\sum_{i=1}^{N} K_i \,.$$

Serfling's inequality

- **For a set of N random variables K_i with values $k_i \in \{0,1\}$, where $i \in \{1,\cdots,N\}$, the average is defined as**

- $$K = \frac{1}{N}\sum_{i=1}^{N} K_i \,.$$ **Average of string**

- **Suppose we draw a sample (without replacement) of size n out of the set $\{K_i\}_i$ with values $x_j \in \{0,1\}$, where $j \in \{1,\cdots,n\}$. Then its average is defined as**

- $$X = \frac{1}{n}\sum_{j=1}^{n} X_j \,.$$ **Average of sample**

Serfling's inequality

- **For a set of N random variables K_i with values $k_i \in \{0,1\}$, where $i \in \{1,\cdots,N\}$, the average is defined as**

$$K = \frac{1}{N}\sum_{i=1}^{N} K_i \,.$$

- **Suppose we draw a sample (without replacement) of size n out of the set $\{K_i\}_i$ with values $x_j \in \{0,1\}$, where $j \in \{1,\cdots,n\}$. Then its average is defined as**

$$X = \frac{1}{n}\sum_{j=1}^{n} X_j \,.$$

- **Now let** $k = N - n$, and $0 \le \beta \le 1$. Then

$$\Pr[X \ge K + \beta] \le e^{-\frac{2\beta^2 nN}{k+1}}.$$

Parameter estimation

Serfling's inequality in brief:

The probability that the sample average X is bigger than the total average K is exponentially small in the sample size.

Overview of the proof of Serfling's inequality

Assume a binomial distribution. $B(n, p)$

$$X_i = \begin{cases} 1 & \text{if the } i\text{th trial is a success}. \\ 0 & \text{otherwise} \end{cases}$$

Overview of the proof of Serling's inequality

Assume a binomial distribution. $B(n, p)$

$$X_i = \begin{cases} 1 & \text{if the } i\text{th trial is a success.} \\ 0 & \text{otherwise} \end{cases}$$

$$X = \sum_{i \in [n]} X_i$$

Write its characteristic function.

$$E(e^{\lambda X}) = E\Big[\sum_{i \geq 0} \frac{\lambda^i}{i!} X^i\Big] = \sum_{i \geq 0} \frac{\lambda^i}{i!} E[X^i]$$

$$E[e^{\lambda X}] = E[e^{\lambda \sum_i X_i}] = E\Big[\Pi_i e^{\lambda X_i}\Big] = \Pi_i E[e^{\lambda X_i}] = (pe^{\lambda} + q)^n$$

Overview of the proof of Serling's inequality

Assume a binomial distribution. $B(n, p)$

$$X_i = \begin{cases} 1 & \text{if the } i\text{th trial is a success.} \\ 0 & \text{otherwise} \end{cases}$$

Write its characteristic function.

$$E(e^{\lambda X}) = E\Big[\sum_{i\geq 0} \frac{\lambda^i}{i!} X^i\Big] = \sum_{i\geq 0} \frac{\lambda^i}{i!} E[X^i]$$

$$E[e^{\lambda X}] = E[e^{\lambda \sum_i X_i}] = E\Big[\Pi_i e^{\lambda X_i}\Big] = \Pi_i E[e^{\lambda X_i}] = (pe^{\lambda} + q)^n$$

For $\lambda > 0$ $\Pr[X > m] = \Pr[e^{\lambda X} > e^{\lambda m}] \leq \dfrac{E[e^{\lambda X}]}{e^{\lambda m}}$

Find a minima w.r.t. the transform parameter λ.

$$\Pr[X > (p+t)n] \leq \left(\left(\frac{p}{p+t}\right)^{p+t}\left(\frac{q}{q-t}\right)^{q-t}\right)^n$$

$$m = (p+t)n$$

$$e^{\lambda} = \frac{q(p+t)}{p\{1-(p+t)\}}$$

Overview of the proof of Serling's inequality

Assume a binomial distribution. $B(n, p)$

$$X_i = \begin{cases} 1 & \text{if the } i\text{th trial is a success.} \\ 0 & \text{otherwise} \end{cases}$$

Write its characteristic function.

$$E(e^{\lambda X}) = E\Big[\sum_{i\geq 0} \frac{\lambda^i}{i!} X^i\Big] = \sum_{i\geq 0} \frac{\lambda^i}{i!} E[X^i]$$

$$E[e^{\lambda X}] = E[e^{\lambda \sum_i X_i}] = E\Big[\Pi_i e^{\lambda X_i}\Big] = \Pi_i E[e^{\lambda X_i}] = (pe^{\lambda} + q)^n$$

Markov inequality

For $\lambda > 0$ $\Pr[X > m] = \Pr[e^{\lambda X} > e^{\lambda m}] \leq \dfrac{E[e^{\lambda X}]}{e^{\lambda m}}$

Find a minima w.r.t. the transform parameter λ.

$$m = (p + t)n$$

$$e^{\lambda} = \frac{q(p + t)}{p\{1 - (p + t)\}}$$

$$\Pr[X > (p + t)n] \leq \left(\left(\frac{p}{p + t}\right)^{p+t} \left(\frac{q}{q - t}\right)^{q-t}\right)^n$$

Brief overview of Markov inequality

$$E(X) = P(X < a) \cdot E(X \mid X < a) + P(X > a) E(X \mid X > a)$$

$$E(X \mid X < a) \geq 0; \quad E(X \mid X > a) \geq a$$

$$E(X) \geq P(X \geq a) \cdot E(X \mid X \geq a) \geq a \cdot P(X \geq a) \implies P(X \geq a) \leq \frac{E(X)}{a}.$$

Overview of the proof of Serling's inequality

$$\Pr[X > (p+t)n] \leq \left(\left(\frac{p}{p+t} \right)^{p+t} \left(\frac{q}{q-t} \right)^{q-t} \right)^n$$

$$= \exp\left(-(p+t)\ln\frac{p+t}{p} - (q-t)\ln\frac{q-t}{q} \right)^n$$

$$= \exp\left(D\{(p+t, q-t) \,||\, (p,q)\} \right)^n$$

$$f(t) := (p+t)\ln\frac{p+t}{p} + (q-t)\ln\frac{q-t}{q}$$

$$f'(t) = \ln\frac{p+t}{p} - \ln\frac{q-t}{q}$$

$$f''(t) = \frac{1}{(p+t)(q-t)}$$

$$f(0) = 0 = f'(0) \quad f''(t) \geq 4 \text{ for all } 0 \leq t \leq q$$

Because $xy \leq \frac{1}{4}$ For any two nonnegative real numbers adding upto 1.

$$f(t) = f(0) + f'(0)t + f''(\xi)\frac{t^2}{2!}, \qquad 0 < \xi < t$$
$$\geq 2t^2$$

$$\Pr[X > E(X) + t], \Pr[X < E(X) - t] \leq e^{-2t^2/n}$$

Parameter estimation

$\Lambda_n =$ Error rate in the remaining n bits

$\Lambda_k =$ Error rate in the k sample bits

$\lambda_{\max} =$ Threshold for the sample error rate

$\gamma =$ Small constant

$$\Lambda_k \leq \lambda_{\max}$$

Parameter estimation

$\Lambda_n =$ Error rate in the remaining n bits

$\Lambda_k =$ Error rate in the k sample bits

$\lambda_{\max} =$ Threshold for the sample error rate

$\gamma =$ Small constant

$$\Lambda_k \leq \lambda_{\max}$$

Bound on error rate

Parameter estimation

$\Lambda_n =$ Error rate in the remaining n bits

$\Lambda_k =$ Error rate in the k sample bits

$\lambda_{\max} =$ Threshold for the sample error rate

$\gamma =$ Small constant

$$\Lambda_k \leq \lambda_{\max}$$

Bound on error rate

Quantity of interest

$$\Pr\left[\Lambda_n \geq \Lambda_k + \gamma \,\middle|\, \Lambda_k \leq \lambda_{\max}\right] = ???$$

Parameter estimation

$\Lambda_n =$ Error rate in the remaining n bits

$\Lambda_k =$ Error rate in the k sample bits

$\lambda_{\max} =$ Threshold for the sample error rate

$\gamma =$ Small constant

$$\Lambda_k \leq \lambda_{\max}$$

Bound on error rate

Quantity of interest

$$\Pr\left[\Lambda_n \geq \Lambda_k + \gamma \,\middle|\, \Lambda_k \leq \lambda_{\max}\right] = ???$$

(Probability for the error rate in the remaining n bits is greater than the error rate in the k sample bits plus a small constant γ given that λ_k is bounded by a constant threshold $\lambda_{\max}$ for the sample error rate)

Parameter estimation

K_A : Alice's string

K_B : Bob's string

K_A^k **and** K_B^k : Alice and Bob's sampled bits

K_A^n **and** K_B^n : Alice and Bob's remaining bits

Parameter estimation

$\boldsymbol{K_A}$: Alice's string

$\boldsymbol{K_B}$: Bob's string

$\boldsymbol{K_A^k}$ **and** $\boldsymbol{K_B^k}$: Alice and Bob's sampled bits

$\boldsymbol{K_A^n}$ **and** $\boldsymbol{K_B^n}$: Alice and Bob's remaining bits

$$K_A = K_A^k K_A^n, \qquad K_B = K_B^k K_B^n$$

Parameter estimation

$\boldsymbol{K_A}$: Alice's string

$\boldsymbol{K_B}$: Bob's string

$\boldsymbol{K_A^k}$ **and** $\boldsymbol{K_B^k}$: Alice and Bob's sampled bits

$\boldsymbol{K_A^n}$ **and** $\boldsymbol{K_B^n}$: Alice and Bob's remaining bits

$$K_A = K_A^k K_A^n, \qquad K_B = K_B^k K_B^n$$

$$K_A^k \oplus K_B^k = 0 \implies \text{Alice and Bob have same bits.}$$
$$K_A^k \oplus K_B^k = 1 \implies \text{Alice and Bob's bits do not match.}$$

- Error rates

$$\Lambda_k = \frac{1}{k} | K_A^k \oplus K_B^k |$$

$$\Lambda_n = \frac{1}{n} | K_A^n \oplus K_B^n |$$

Hamming weight=# 1 in the string

- Error rates

$$\Lambda_k = \frac{1}{k} |K_A^k \oplus K_B^k|$$

$$\Lambda_n = \frac{1}{n} |K_A^n \oplus K_B^n|$$

Hamming weight=# 1 in the string

Define $\nu = \frac{k}{N}$

$$\Lambda = \frac{1}{N} |K_A \oplus K_B| = \nu \Lambda_k + (1 - \nu) \Lambda_n$$

- Error rates

$$\Lambda_k = \frac{1}{k} |K_A^k \oplus K_B^k|$$

$$\Lambda_n = \frac{1}{n} |K_A^n \oplus K_B^n|$$

Hamming weight=# 1 in the string

Define $\nu = \frac{k}{N}$

$$\boldsymbol{\Lambda = \frac{1}{N} |K_A \oplus K_B| = \nu \Lambda_k + (1 - \nu) \Lambda_n}$$

Bayes' theorem

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

$$\Pr[\Lambda_n \geq \Lambda_k + \gamma \,|\, \Lambda_k \leq \lambda_{\max}] \leq \frac{\Pr[\Lambda_n \geq \Lambda_k + \gamma]}{\Pr[\Lambda_k \leq \lambda_{\max}]}$$

$$\Pr[\Lambda_n \geq \Lambda_k + \gamma \,|\, \Lambda_k \leq \lambda_{\max}] \leq \frac{\Pr[\Lambda_n \geq \Lambda_k + \gamma]}{\Pr[\Lambda_k \leq \lambda_{\max}]}$$

Probability that the protocol is not aborted.

???

$$\Pr[\Lambda_n \geq \Lambda_k + \gamma \,|\, \Lambda_k \leq \lambda_{\max}] \leq \frac{\Pr[\Lambda_n \geq \Lambda_k + \gamma]}{\Pr[\Lambda_k \leq \lambda_{\max}]}$$

???

$$\Pr[\Lambda_n \geq \Lambda_k + \gamma \,|\, \Lambda_k \leq \lambda_{\max}] \leq \frac{\Pr[\Lambda_n \geq \Lambda_k + \gamma]}{\Pr[\Lambda_k \leq \lambda_{\max}]}$$

$$\Pr[\Lambda_n \geq \Lambda_k + \gamma] = \Pr[\nu\Lambda_n \geq \nu\Lambda_k + \nu\gamma]$$

$$\Pr[\Lambda_n \geq \Lambda_k + \gamma] = \Pr[\nu\Lambda_n \geq \nu\Lambda_k + \nu\gamma]$$

$$= \Pr[\Lambda_n \geq \nu\Lambda_k + (1-\nu)\Lambda_n + \nu\gamma]$$

$$\Pr[\Lambda_n \geq \Lambda_k + \gamma] = \Pr[\nu\Lambda_n \geq \nu\Lambda_k + \nu\gamma]$$

$$= \Pr[\Lambda_n \geq \nu\Lambda_k + (1-\nu)\Lambda_n + \nu\gamma]$$

$$= \Pr[\Lambda_n \geq \Lambda + \nu\gamma]$$

$$\Pr[\Lambda_n \geq \Lambda_k + \gamma] = \Pr[\nu\Lambda_n \geq \nu\Lambda_k + \nu\gamma]$$

$$= \Pr[\Lambda_n \geq \nu\Lambda_k + (1-\nu)\Lambda_n + \nu\gamma]$$

$$= \Pr[\Lambda_n \geq \Lambda + \nu\gamma]$$

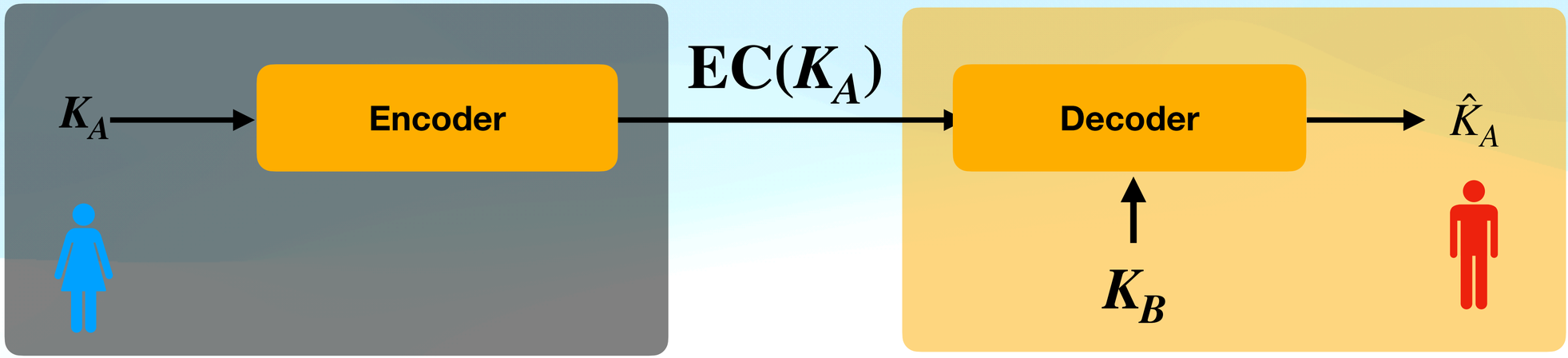
$$\leq e^{-\frac{2k^2 n\gamma^2}{(k+1)N}}$$

$$\Pr[\Lambda_n \geq \Lambda_k + \gamma \,|\, \Lambda_k \leq \lambda_{\max}] \leq \frac{\Pr[\Lambda_n \geq \Lambda_k + \gamma]}{\Pr[\Lambda_k \leq \lambda_{\max}]}$$

$$\Pr[\Lambda_n \geq \Lambda_k + \gamma \,|\, \Lambda_k \leq \lambda_{\max}] \leq \frac{e^{-\frac{2k^2 n\gamma^2}{(k+1)N}}}{p_{\text{pass}}}$$

Error correction

Goal: Make Alice and Bob's strings equal.



Classical error correction: Well studied problem
Independent of protocol: Check if it was successful

Step II: Error correction

Error Correction

- We know error rate, but how to correct errors?

Example: **XOR operation**

1. Alice randomly choose two numbers and computes their XOR.
2. She sends the output and tells the position of bits to Bob.
3. If Bob gets a different output, they discard both bits.
4. If they get the same output, they keep first bit and discard the other.

Error Correction

- We know error rate, but how to correct errors?

Example: **XOR operation**

1. Alice randomly choose two numbers and computes their XOR.
2. She sends the output and tells the position of bits to Bob.
3. If Bob gets a different output, they discard both bits.
4. If they get the same output, they keep first bit and discard the other.

Not efficient

There exist efficient error correction protocols

To be seen

Error Correction

But, given error correction is done, **how to figure out whether it is successful or not?**

Error Correction

But, given error correction is done, how to figure out whether it is successful or not?

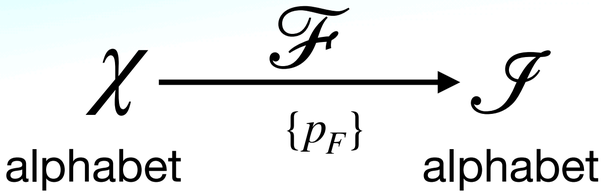
Solution: Two-universal hash function

Error Correction

But, given error correction is done, how to figure out whether it is successful or not?

Solution: Two-universal hash function

Let $\mathcal{F}$ be a family of functions from an alphabet $\mathcal{X}$ to an alphabet $\mathcal{Z}$ and let p_F be a probability distribution on $\mathcal{F}$. The pair $(\mathcal{F}, p_F)$ is called two-universal if



The pair $(\mathcal{F}, p_F)$ is called two-universal if

$$\Pr_{f \in \mathcal{F}}[f(x) = f(x')] \leq \frac{1}{|\mathcal{I}|}$$

For any $x, x' \in \chi$ with $x \neq x'$ and f chosen randomlyfrom $\mathcal{F}$ according to p_F.

Checking Procedure

1. Alice randomly choose a function from a family of two-universal hash functions and applies it to her bit string.
2. She sends both the function f_{EC} she chose and the output $f_{\mathrm{EC}}(K_A)$ to Bob.
3. Bob evaluates function on his key and obtains $f_{\mathrm{EC}}(K_B)$.
4. He compares his rest to Alice's output.

If their hashes are equal, their keys are same with high probability!

Or abort the protocol

How it is ensured that protocol is correct with two-universal hash function?

If the cardinality of the output space is $|\mathcal{J}| = 2^{\left\lceil \log \frac{1}{\epsilon_{\text{cor}}} \right\rceil}$, keys are equal except with probability ϵ_{cor}

Probability with which the two string differ from each other

How it is ensured that protocol is correct with two-universal hash function?

If the cardinality of the output space is $|\mathcal{J}| = 2^{\lceil \log \frac{1}{\epsilon_{\text{cor}}} \rceil}$, keys are equal except with probability ϵ_{cor}

Probability with which the two string differ from each other

$$\Pr\left[f_{\text{EC}}(K_A)] = f_{\text{EC}}(K_B) \middle| K_A \neq K_B\right] \leq 2^{-\lceil \log \frac{1}{\epsilon_{\text{cor}}} \rceil} \leq \epsilon_{\text{cor}}$$

$$\Pr\left[f_{\mathrm{EC}}(K_A) = f_{\mathrm{EC}}(K_B) \,\middle|\, K_A \neq K_B\right] \leq 2^{-[\log \frac{1}{\epsilon_{\mathrm{cor}}}]} \leq \epsilon_{\mathrm{cor}}$$

Together with Bayes theorem

$$\underbrace{\Pr\left[f_{\mathrm{EC}}(K_A) = f_{\mathrm{EC}}(K_B) \,\middle|\, K_A \neq K_B\right]}_{\leq \epsilon_{\mathrm{cor}}} \underbrace{\Pr\left[K_A \neq K_B\right]}_{\leq 1} = \Pr\left[K_A \neq K_B \,\middle|\, f_{\mathrm{EC}}(K_A) = f_{\mathrm{EC}}(K_B)\right] \underbrace{\Pr\left[f_{\mathrm{EC}}(K_A) = f_{\mathrm{EC}}(K_B)\right]}_{=1}$$

$$\Pr[K_A \neq K_B \,|\, f_{\mathrm{EC}}(K_A) = f_{\mathrm{EC}}(K_B)] \leq \epsilon_{\mathrm{cor}}$$

This procedure is independent of error rate that Alice and Bob have observed in the parameter estimation step and also independent of the chosen error correcting code.

Step III: Privacy amplification

Privacy Amplification

Remove any knowledge that Eve has of the key after all the other steps in the protocol.

Privacy Amplification

Goal: Remove any knowledge that Eve has of the key after all the other steps in the protocol.

Tool: Randomness extractors

Input: Source of randomness (bit strings)+small uniformly random string (seed)
Output: Almost uniformly random string

Requirements:

1. Output string is independent of seed ⟶ strong randomness extractor
2. Take quantum adversary into account ⟶ quantum-proof strong randomness extractor

Privacy amplification

Alice's system: Classical random variable X

Eve's system: Quantum system E

State of composite system

$$\rho_{XE} = \sum_{x \in \mathcal{X}} p_X(x) \, |x\rangle\langle x| \otimes \rho_E^x$$

$\{ |x\rangle \}$**:** is an ONB of Alice's system.

System of seed: $\rho_Y \in \mathcal{B}(\mathcal{H}_Y)$

Privacy amplification

How to quantify Eve's information?

Definition (Quantum conditional min-entropy)

The quantum conditional min-entropy of a state $\rho_{AB} \in \mathcal{B}(\mathcal{H}_A \otimes \mathcal{H}_B)$, conditioned on system B Is defined as

$$H_{\min}(A|B) = \sup_{\sigma_B} H_{\min}(\rho_{AB}|\sigma_B) = -\log \min_{\sigma_B} \min\{\lambda : \rho_{AB} \leq \lambda \cdot 1_A \otimes \sigma_B\}$$

Where the minimum is over all the states $\sigma_B \in \mathcal{B}(\mathcal{H}_B)$.

Operational interpretation (for cq state)

Amount of uniform randomness that we can extract from a classical random variable that is correlated with a quantum system such that the result is independent of the quantum system.

$(k, \epsilon)-$ strong quantum- proof randomness extractor

A function $\mathrm{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$
if for all classical-quantum states ρ_{XE}
with a classical random variable $X \in \{0,1\}^n$ **with min-entropy**
$H_{\min}(X|E) \geq k$ **and a uniform seed** $Y \in \{0,1\}^d$ **we have**

$$\frac{1}{2}\left|\left|\rho_{\mathrm{Ext(X,Y)YE}} - \frac{1}{2^m} \otimes \rho_Y \otimes \rho_E\right|\right|_1 \leq \epsilon\,.$$

$\frac{\mathbf{1}}{2^m} \otimes \rho_Y \otimes \rho_E$: **ideal situation.**

Independent of seed and Y **and the state of Eve's system** E.