Shor_Preskill simple proof of security of BB84 protocol

Based on

Simple Proof of Security of the BB84 Quantum Key Distribution Protocol

Peter Shor and John Preskill

Phys. Rev. Lett. **85**, 441 – Published 10 July 2000

A detour to CSS (Calderbank-Shor-Steane) Codes as it will be used as a tool.

- *Uhlmann's Theorem*:

$$F(\rho,\sigma)=\max_{\psi,\phi}F(\psi,\phi), \tag{24}$$

where ψ and ϕ are purifications (using the same ancillary system) of ρ and σ respectively.

Proof. Denote the system of ρ and σ as A and the ancillary system as E. Consider the maximally entangled state $|\Phi^+\rangle=\sum_j|jj\rangle$, an explicit purification of ρ or σ is $|\psi\rangle_{AE}=\sqrt{\rho_A}\otimes\mathbb{I}_E|\Phi^+\rangle_{AE}$ or $|\phi\rangle_{AE}=\sqrt{\sigma_A}\otimes\mathbb{I}_E|\Phi^+\rangle_{AE}$. Then an arbitrary purification of ρ or σ is $U_E|\psi\rangle_{AE}$ or $V_E|\phi\rangle_{AE}$. Now we have

$$\begin{aligned}F(\psi,\phi)&=|\langle\phi_{AE}|V_EU_E|\psi_{AE}\rangle|,\\&=|\langle\Phi^+|_{AE}\sqrt{\sigma_A}V_EU_E|\sqrt{\rho_A}|\Phi^+\rangle_{AE}|,\\&=|\mathrm{Tr}[\sqrt{\sigma_A}\sqrt{\rho_A}\cdot V_EU_E|]|.\end{aligned} \tag{25}$$

Here the last line uses the identity $\mathrm{Tr}[A^\dagger B]=\langle\Phi^+|A\otimes B|\Phi^+\rangle$. Using the variational form of the trace norm, we have

$$\max_{\psi,\phi}F(\psi,\phi)=\max_{U_EV_E}|\mathrm{Tr}[\sqrt{\sigma_A}\sqrt{\rho_A}\cdot V_EU_E|]|=|\mathrm{Tr}[|\sqrt{\sigma_A}\sqrt{\rho_A}|]|=F(\rho,\sigma). \tag{26}$$

□

3 Relation between trace distance and fidelity

Pure states For two pure states ψ and ϕ, trace distance and fidelity are actually equivalent

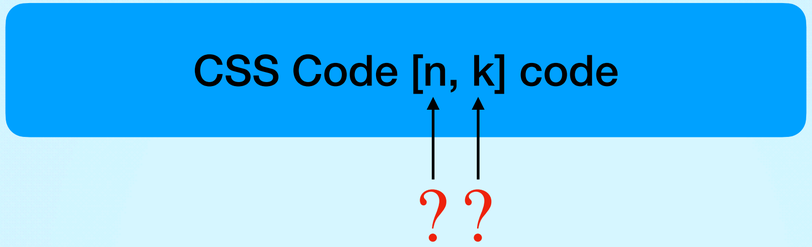
$$D(\psi, \phi) = \sqrt{1 - F(\phi, \psi)^2}. \tag{30}$$

General case In general, we have

$$1 - F(\rho, \sigma) \leq D(\rho, \sigma) \leq \sqrt{1 - F(\rho, \sigma)^2}. \tag{31}$$

The first inequality could be proven by using the reduction to classical trace distance and fidelity and their relation as we proved in Eq. (17). For the second inequality, we consider the purification of ρ and σ, i.e., ψ and ϕ, that achieves the fidelity according to Uhlmann's theorem. Then we have

$$D(\rho, \sigma) \leq D(\psi, \phi) = \sqrt{1 - F(\phi, \psi)^2} = \sqrt{1 - F(\rho, \sigma)^2}. \tag{32}$$



Generator matrix G: maps messages to their equivalents in code spaces.

Code $\rightarrow y = Gx \leftarrow$ Message

CSS Code [n, k] code

Generator matrix G: maps messages to their equivalents in code spaces.

Code $\rightarrow y = Gx \leftarrow$ Message

$0 \rightarrow (0,0,0)^T$ $1 \rightarrow (1,1,1)^T$

CSS Code [n, k] code

Generator matrix G: maps messages to their equivalents in code spaces.

Code $\rightarrow y = Gx \leftarrow$ Message

$0 \rightarrow (0,0,0)^T$ $\quad$ $1 \rightarrow (1,1,1)^T$

$G = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$ $\quad$ $Hy = 0$

$(n-k) \times n$ Matrix with entries in $\{0,1\}$

CSS Code [n, k] code

Generator matrix G: maps messages to their equivalents in code spaces.

Code $\rightarrow y = Gx \leftarrow$ Message

$0 \rightarrow (0,0,0)^T$ $\quad$ $1 \rightarrow (1,1,1)^T$

$G = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$ $\quad$ $Hy = 0$

$(n-k) \times n$ Matrix with entries in $\{0,1\}$

$(n-k)$ Linearly independent vectors orthogonal to the columns of G.

$$H = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}$$

error $e \implies y' \to y + e$

Since $Hy = 0$ for all codewords $y \implies$

$Hy' = Hy + He = He \leftarrow$ Error syndrome

error $e \implies y' \to y + e$

Since $Hy = 0$ for all codewords $y \implies$

$Hy' = Hy + He = He \leftarrow$ Error syndrome

Different error syndromes

$$He_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad He_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad He_3 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

error $e \implies y' \to y + e$

Since $Hy = 0$ for all codewords $y \implies$

$$Hy' = Hy + He = He \leftarrow \text{ Error syndrome}$$

Different error syndromes

$$He_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad He_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad He_3 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Makes it possible to read off the error from syndromes.

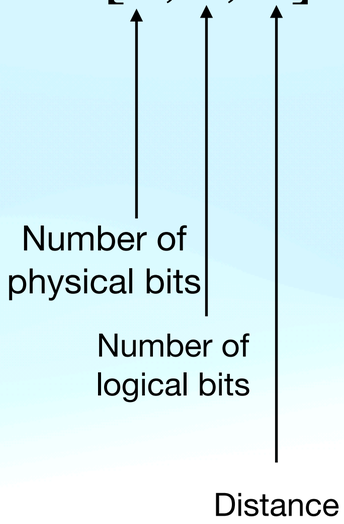
$d(x, y) \to$ Hamming distance

The number of positions in which the two bit strings differ

How many errors can such a code correct?

$$d(C) = \min_{x,y \in C, x \neq y} d(x, y)$$

$$C \rightarrow \text{An } [n, k, d] \text{ code}$$



How many errors can such a code correct?

$$d(C) = \min_{x,y \in C, x \neq y} d(x, y)$$

$$C \rightarrow \text{An } [n, k, d] \text{ code}$$

A code with distance $(2t + 1)$ for some integer $'t'$ can be used to correct upto $'t'$ errors, simply by decoding the corrupted message $'y'$ as the unique codeword y that satisfies

$$d(y, y') \leq t\,.$$

Duality

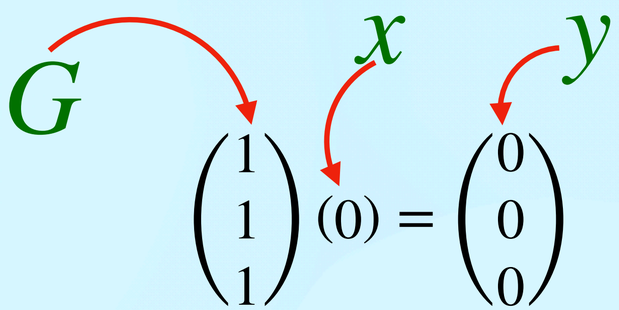
C : $[n, k]$ Code with generator matrix G and parity check matrix H .

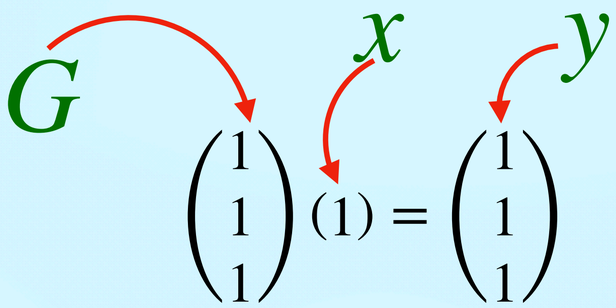
$C^{\perp}$: Codeword orthogonal to each codeword in C

H^T : Generator matrix of the dual.

G^T : Parity check matrix of the dual.

Example: A [3, 1, 1] code



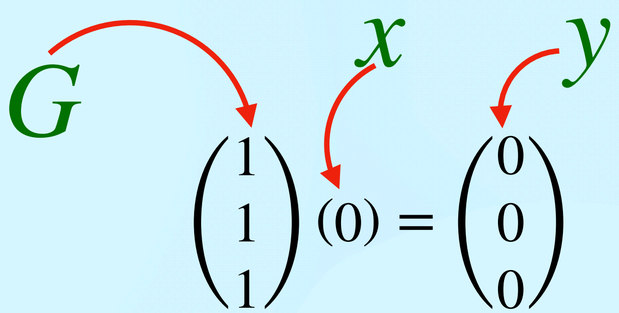


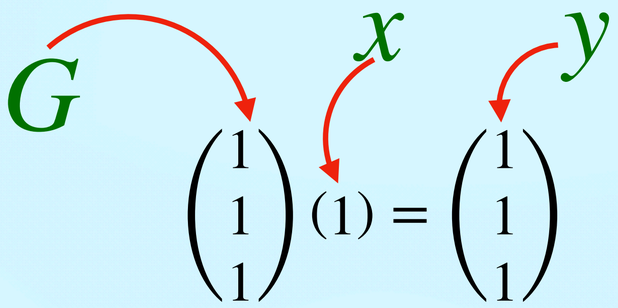
$$\begin{pmatrix}0\\0\\0\end{pmatrix} \rightarrow \begin{pmatrix}1\\0\\0\end{pmatrix}$$

$$\begin{pmatrix}1\\1\\1\end{pmatrix} \rightarrow \begin{pmatrix}0\\1\\1\end{pmatrix}$$

1st bit corrupted

Example: A [3, 1, 1] code





$$\begin{pmatrix}0\\0\\0\end{pmatrix} \to \begin{pmatrix}1\\0\\0\end{pmatrix};$$

$$\begin{pmatrix}1\\1\\1\end{pmatrix} \to \begin{pmatrix}0\\1\\1\end{pmatrix};$$

1st bit corrupted

$$\begin{pmatrix}0\\0\\0\end{pmatrix} \to \begin{pmatrix}0\\1\\0\end{pmatrix}$$

$$\begin{pmatrix}1\\1\\1\end{pmatrix} \to \begin{pmatrix}1\\0\\1\end{pmatrix}$$

2nd bit corrupted

Example: A [3, 1, 1] code

$$G \; x \; y \qquad \begin{pmatrix}1\\1\\1\end{pmatrix}(0) = \begin{pmatrix}0\\0\\0\end{pmatrix}$$

$$G \; x \; y \qquad \begin{pmatrix}1\\1\\1\end{pmatrix}(1) = \begin{pmatrix}1\\1\\1\end{pmatrix}$$

$$\begin{pmatrix}0\\0\\0\end{pmatrix} \to \begin{pmatrix}1\\0\\0\end{pmatrix}; \qquad \begin{pmatrix}1\\1\\1\end{pmatrix} \to \begin{pmatrix}0\\1\\1\end{pmatrix};$$

1st bit corrupted

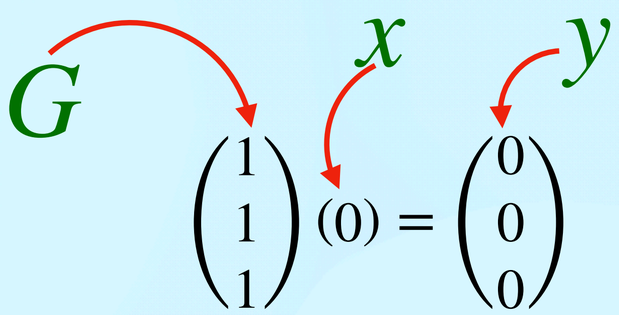
$$\begin{pmatrix}0\\0\\0\end{pmatrix} \to \begin{pmatrix}0\\1\\0\end{pmatrix}; \qquad \begin{pmatrix}1\\1\\1\end{pmatrix} \to \begin{pmatrix}1\\0\\1\end{pmatrix};$$

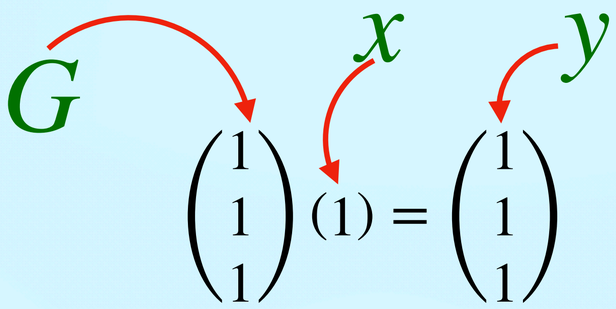
2nd bit corrupted

$$\begin{pmatrix}0\\0\\0\end{pmatrix} \to \begin{pmatrix}0\\0\\1\end{pmatrix}; \qquad \begin{pmatrix}1\\1\\1\end{pmatrix} \to \begin{pmatrix}1\\1\\0\end{pmatrix};$$

3rd bit corrupted

Example: A [3, 1, 1] code





$\begin{pmatrix}0\\0\\0\end{pmatrix} \to \begin{pmatrix}1\\0\\0\end{pmatrix}$;

$\begin{pmatrix}1\\1\\1\end{pmatrix} \to \begin{pmatrix}0\\1\\1\end{pmatrix}$;

1st bit corrupted

$\begin{pmatrix}0\\0\\0\end{pmatrix} \to \begin{pmatrix}0\\1\\0\end{pmatrix}$;

$\begin{pmatrix}1\\1\\1\end{pmatrix} \to \begin{pmatrix}1\\0\\1\end{pmatrix}$;

2nd bit corrupted

$\begin{pmatrix}0\\0\\0\end{pmatrix} \to \begin{pmatrix}0\\0\\1\end{pmatrix}$;

$\begin{pmatrix}1\\1\\1\end{pmatrix} \to \begin{pmatrix}1\\1\\0\end{pmatrix}$;

3rd bit corrupted

$\begin{pmatrix}0\\0\\0\end{pmatrix} \to \begin{pmatrix}0\\0\\0\end{pmatrix}$

$\begin{pmatrix}1\\1\\1\end{pmatrix} \to \begin{pmatrix}1\\1\\1\end{pmatrix}$

No error

$$H = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}$$

$$\underbrace{\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}}_{H} \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad \underbrace{\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}}_{H} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \longleftarrow$$

NO Error.

$$H = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}$$

$$\underbrace{\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}}_{H} \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad \underbrace{\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}}_{H} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \longleftarrow$$

NO Error.

$$\underbrace{\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}}_{H} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad \underbrace{\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}}_{H} \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \longleftarrow$$

Flip on the first qubit.

$$H = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}$$

$$\underbrace{\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}}_{H} \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad \underbrace{\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}}_{H} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \longleftarrow$$ NO Error.

$$\underbrace{\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}}_{H} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad \underbrace{\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}}_{H} \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \longleftarrow$$ Flip on the first bit.

$$\underbrace{\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}}_{H} \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \qquad \underbrace{\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}}_{H} \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \longleftarrow$$ Flip on the second bit.

$$H = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}$$

NO Error.

$$\underbrace{\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}}_{H} \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad \underbrace{\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}}_{H} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\underbrace{\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}}_{H} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \qquad \underbrace{\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}}_{H} \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Flip on the third bit.

$$\underbrace{\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}}_{H} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad \underbrace{\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}}_{H} \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

Flip on the first bit.

$$\underbrace{\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}}_{H} \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \qquad \underbrace{\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}}_{H} \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Flip on the second bit.

Quantum error correction

An $[n, k_1]$ code C_1 An $[n, k_2]$ code C_2 $C_2 \subset C_1$??

C_1 and $C_2^{\perp}$? Both can correct upto t errors.

For any codeword $x \in C_1$, we define the quantum state,

$$|x + C_2\rangle = \frac{1}{\sqrt{|C_2|}} \sum_{y \in C_2} |x + y\rangle$$

$|C_2|$: Cardinality of C_2

Crucial point: Bit flip and phase flip errors are corrected independent of each other.

Bit flip $|x\rangle \rightarrow |x + e_{\text{bit}}\rangle$

Phase flip $|x\rangle \rightarrow (-1)^{x \cdot e_{\text{phase}}} |x\rangle$

Bit flip $|x\rangle \to |x + e_{\text{bit}}\rangle$

Phase flip $|x\rangle \to (-1)^{x \cdot e_{\text{phase}}} |x\rangle$

Observation: Bit flip error $\xrightarrow{\text{Hadamard}}$ Phase flip error

$$|x'\rangle \to |x' + e_{\text{phase}}\rangle$$

$$|x + e_2\rangle \to \frac{1}{\sqrt{2^{k+1}}} \sum_{y \in C_2} (-1)^{(x+y) \cdot e_{\text{phase}}} |x + y + e_{\text{bit}}\rangle$$

Bit flip $|x\rangle \to |x + e_{\text{bit}}\rangle$

Phase flip $|x\rangle \to (-1)^{x \cdot e_{\text{phase}}} |x\rangle$

Observation: Bit flip error $\xrightarrow{\text{Hadamard}}$ Phase flip error

$$|x'\rangle \to |x' + e_{\text{phase}}\rangle$$

$$|x + e_2\rangle \to \frac{1}{\sqrt{2^{k+1}}} \sum_{y \in C_2} (-1)^{(x+y) \cdot e_{\text{phase}}} |x + y + e_{\text{bit}}\rangle$$

Computation of the syndrome: Apply the parity check matrix H_1 of the code C_1 and store the result in the Ancilla state:

$$|x + y + e_{\text{bit}}\rangle \to |x + y + e_{\text{bit}}\rangle |H_1(x + y + e_{\text{bit}})\rangle = |x + y + e_{\text{bit}}\rangle |H_1 e_{\text{bit}}\rangle$$

Bit flip $|x\rangle \rightarrow |x + e_{\text{bit}}\rangle$

Phase flip $|x\rangle \rightarrow (-1)^{x \cdot e_{\text{phase}}} |x\rangle$

Observation: Bit flip error $\xrightarrow{\text{Hadamard}}$ Phase flip error

$$|x'\rangle \rightarrow |x' + e_{\text{phase}}\rangle$$

$$|x + e_2\rangle \rightarrow \frac{1}{\sqrt{2^{k+1}}} \sum_{y \in C_2} (-1)^{(x+y) \cdot e_{\text{phase}}} |x + y + e_{\text{bit}}\rangle$$

Computation of the syndrome: Apply the parity check matrix H_1 of the code C_1 and store the result in the Ancilla state:

$$|x + y + e_{\text{bit}}\rangle \rightarrow |x + y + e_{\text{bit}}\rangle |H_1(x + y + e_{\text{bit}})\rangle = |x + y + e_{\text{bit}}\rangle |H_1 e_{\text{bit}}\rangle$$

Detection of error: Measure the Ancilla state, discard it and apply NOT gate on the qubits where a bit flip has occurred.

This removes all the bit flip errors and the resulting state is:

$$\frac{1}{\sqrt{2^{k_2}}} \sum_{y \in C_2} (-1)^{(x+y)\cdot e_{\text{phase}}} |x+y\rangle$$

This removes all the bit flip errors and the resulting state is:

$$\frac{1}{\sqrt{2^{k_2}}} \sum_{y \in C_2} (-1)^{(x+y)\cdot e_{\text{phase}}} |x+y\rangle$$

$$\frac{1}{\sqrt{2^{k_2}}} \sum_{y \in C_2} (-1)^{(x+y)\cdot e_{\text{phase}}} |x+y\rangle \xrightarrow{H} \frac{1}{\sqrt{2^{n+k_2}}} \sum_{z} \sum_{y \in C_2} (-1)^{(x+y)\cdot (e_{\text{phase}}+z)} |z\rangle$$

This removes all the bit flip errors and the resulting state is:

$$\frac{1}{\sqrt{2^{k_2}}}\sum_{y\in C_2}(-1)^{(x+y)\cdot e_{\text{phase}}}|x+y\rangle$$

$$\frac{1}{\sqrt{2^{k_2}}}\sum_{y\in C_2}(-1)^{(x+y)\cdot e_{\text{phase}}}|x+y\rangle \xrightarrow{H} \frac{1}{\sqrt{2^{n+k_2}}}\sum_{z}\sum_{y\in C_2}(-1)^{(x+y)\cdot(e_{\text{phase}}+z)}|z\rangle$$

A property

(I) If $z' \in C^{\perp}$, $\sum_{y\in C_2}(-1)^{y\cdot z'} = |C_2|$,

(II) if $z' \notin C^{\perp}$, $\sum_{y\in C_2}(-1)^{y\cdot z'} = 0$.

So,

$$\frac{1}{\sqrt{2^{n-k_2}}}\sum_{z'\in C_2^{\perp}}(-1)^{x\cdot z'}|z'+e_{\text{phase}}\rangle$$

This removes all the bit flip errors and the resulting state is:

$$\frac{1}{\sqrt{2^{k_2}}}\sum_{y\in C_2}(-1)^{(x+y)\cdot e_{\text{phase}}}|x+y\rangle$$

$$\frac{1}{\sqrt{2^{k_2}}}\sum_{y\in C_2}(-1)^{(x+y)\cdot e_{\text{phase}}}|x+y\rangle \xrightarrow{H} \frac{1}{\sqrt{2^{n+k_2}}}\sum_{z}\sum_{y\in C_2}(-1)^{(x+y)\cdot(e_{\text{phase}}+z)}|z\rangle$$

A property

(I) If $z'\in C^{\perp}$, $\sum_{y\in C_2}(-1)^{y\cdot z'}=|C_2|$,

(II) (II) if $z'\notin C^{\perp}$, $\sum_{y\in C_2}(-1)^{y\cdot z'}=0.$

For this reason, $C^{\perp}$ is needed.

So,

$$\frac{1}{\sqrt{2^{n-k_2}}}\sum_{z'\in C_2^{\perp}}(-1)^{x\cdot z'}|z'+e_{\text{phase}}\rangle$$

This removes all the bit flip errors and the resulting state is:

$$\frac{1}{\sqrt{2^{k_2}}}\sum_{y\in C_2}(-1)^{(x+y)\cdot e_{\text{phase}}}|x+y\rangle$$

$$\frac{1}{\sqrt{2^{k_2}}}\sum_{y\in C_2}(-1)^{(x+y)\cdot e_{\text{phase}}}|x+y\rangle \xrightarrow{H} \frac{1}{\sqrt{2^{n+k_2}}}\sum_{z}\sum_{y\in C_2}(-1)^{(x+y)\cdot(e_{\text{phase}}+z)}|z\rangle$$

If $z'\in C^{\perp}$, then $\sum_{y\in C_2}(-1)^{y\cdot z'}=|C_2|$, while if $z'\notin C^{\perp}$, $\sum_{y\in C_2}(-1)^{y\cdot z'}=0$. So,

$$\frac{1}{\sqrt{2^{n-k_2}}}\sum_{z'\in C_2^{\perp}}(-1)^{x\cdot z'}|z'+e_{\text{phase}}\rangle$$

Exactly the same form as that of a bit-flip error described the vector e_{phase}.

Example

$$|0\rangle_L \equiv \frac{1}{2\sqrt{2}}\big(|000\rangle + |111\rangle\big)\big(|000\rangle + |111\rangle\big)\big(|000\rangle + |111\rangle\big)$$

$$|1\rangle_L \equiv \frac{1}{2\sqrt{2}}\big(|000\rangle - |111\rangle\big)\big(|000\rangle - |111\rangle\big)\big(|000\rangle - |111\rangle\big)$$

Can correct arbitrary errors on a single qubit.

$$|0\rangle_{L_2} \equiv \frac{1}{2\sqrt{2}}\Big(|0\rangle_{L_1} + |1\rangle_{L_1}\Big)^{\otimes 3}; \quad |1\rangle_{L_2} \equiv \frac{1}{2\sqrt{2}}\Big(|0\rangle_{L_1} - |1\rangle_{L_1}\Big)^{\otimes 3}$$

$$|0\rangle_{L_1} \equiv |000\rangle; \qquad |1\rangle_{L_1} \equiv |111\rangle\,.$$

Shor-Preskill proof for security of BB84 protocol

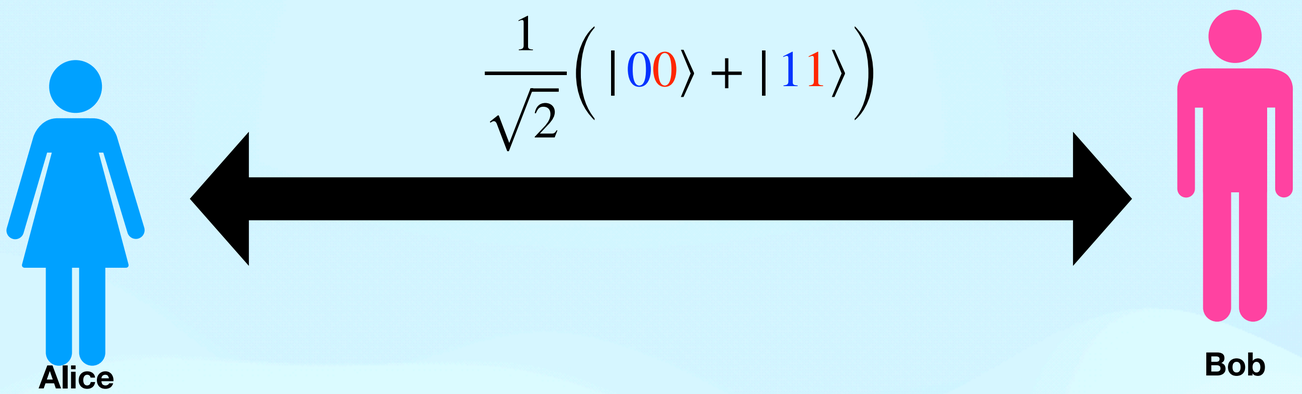
Shor-Preskill proof for security of BB84 protocol

Key result: Key generation rate: $1 - 2h(\epsilon)$

Shor-Preskill proof for security of BB84 protocol

Key result: Key generation rate: $1 - 2h(\epsilon)$

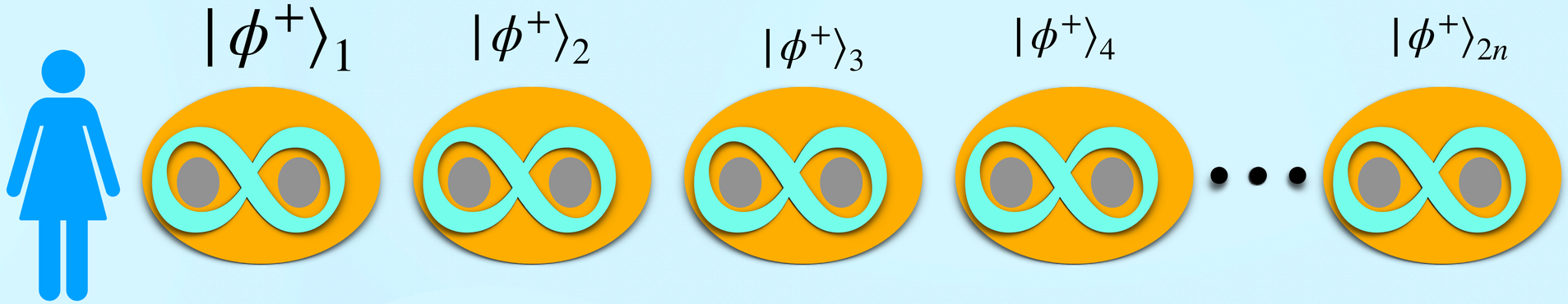
Entanglement-based version of BB84 protocol

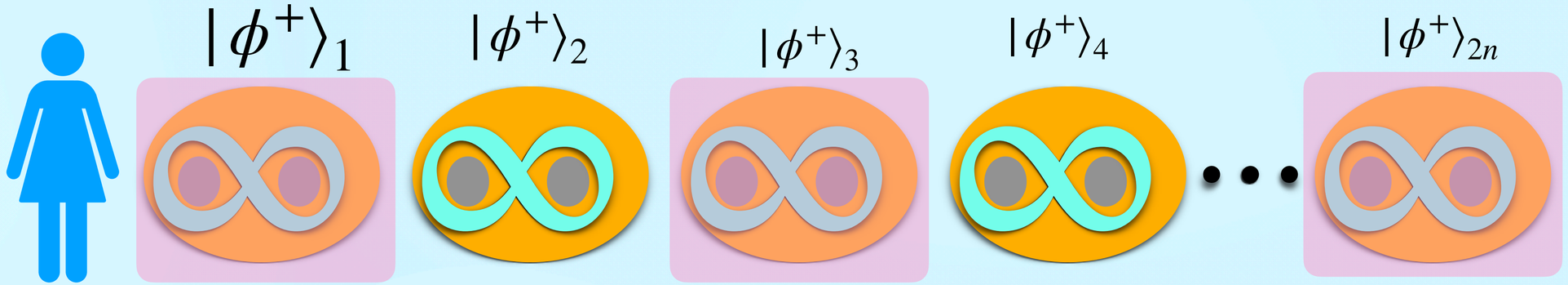


Alice and Bob distribute a sequence of m of these states, i.e.,

$$|\Phi^+\rangle_{AB}^{\otimes m} = |\Phi^+\rangle_{AB} \otimes |\Phi^+\rangle_{AB} \otimes \cdots \otimes |\Phi^+\rangle_{AB}$$

In practice, they will share a mixed state ρ .

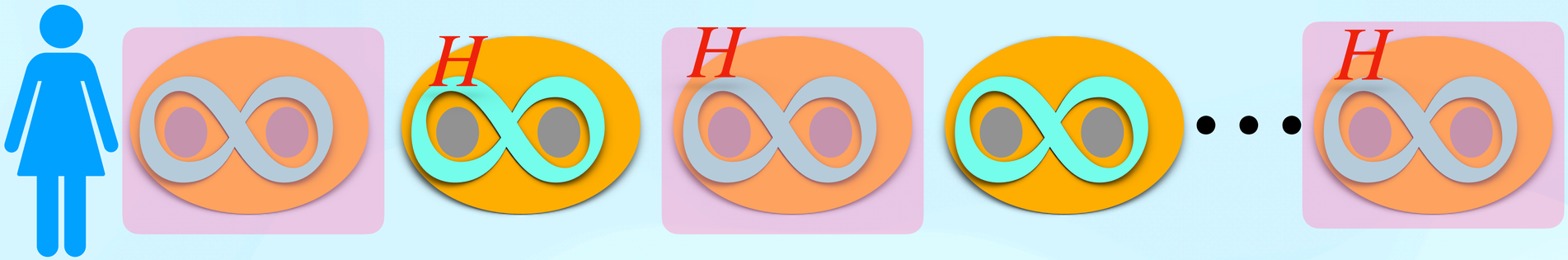

$|\phi^+\rangle_1$
$|\phi^+\rangle_2$
$|\phi^+\rangle_3$
$|\phi^+\rangle_4$
$|\phi^+\rangle_{2n}$

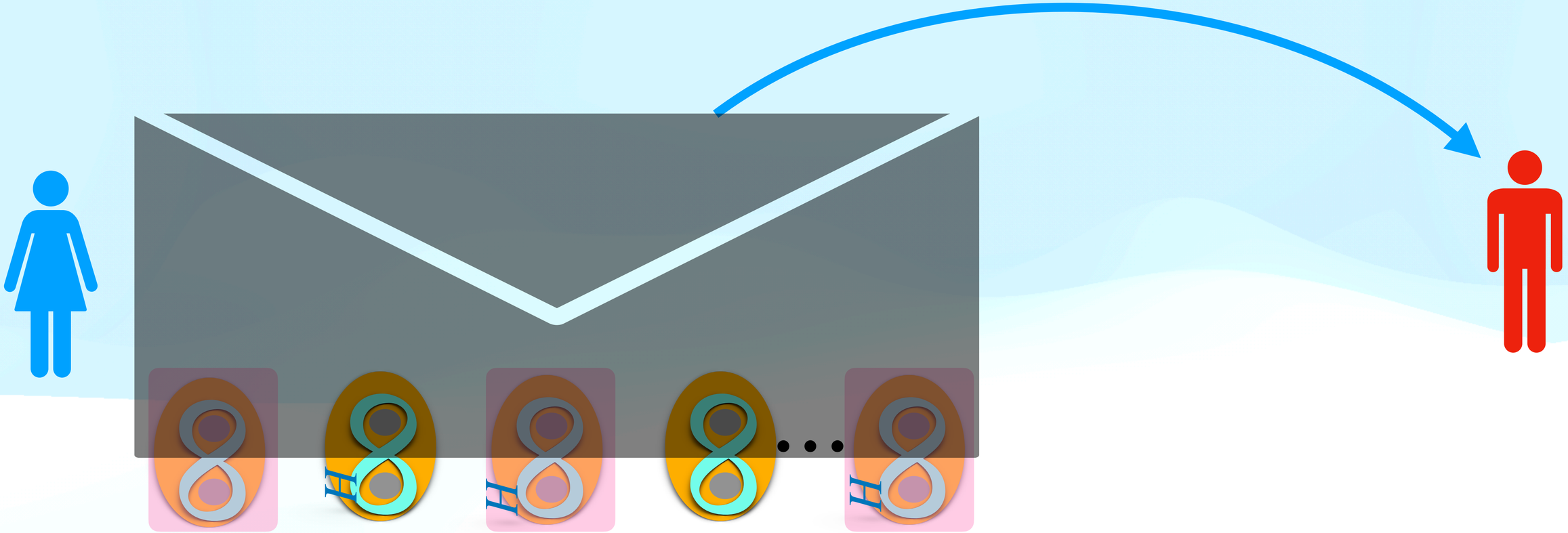

$|\phi^+\rangle_1$
$|\phi^+\rangle_2$
$|\phi^+\rangle_3$
$|\phi^+\rangle_4$
$|\phi^+\rangle_{2n}$

Check bits

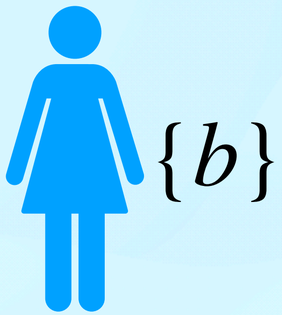
$\{b\}$ 0 1 1 0 1

$\{b\}$ 0 1 1 0 ⋯ 1

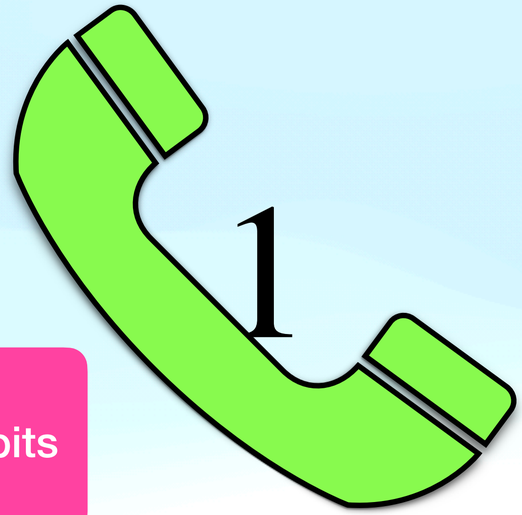




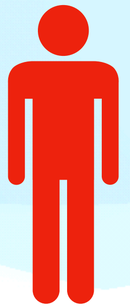
H : Hadamard transformation

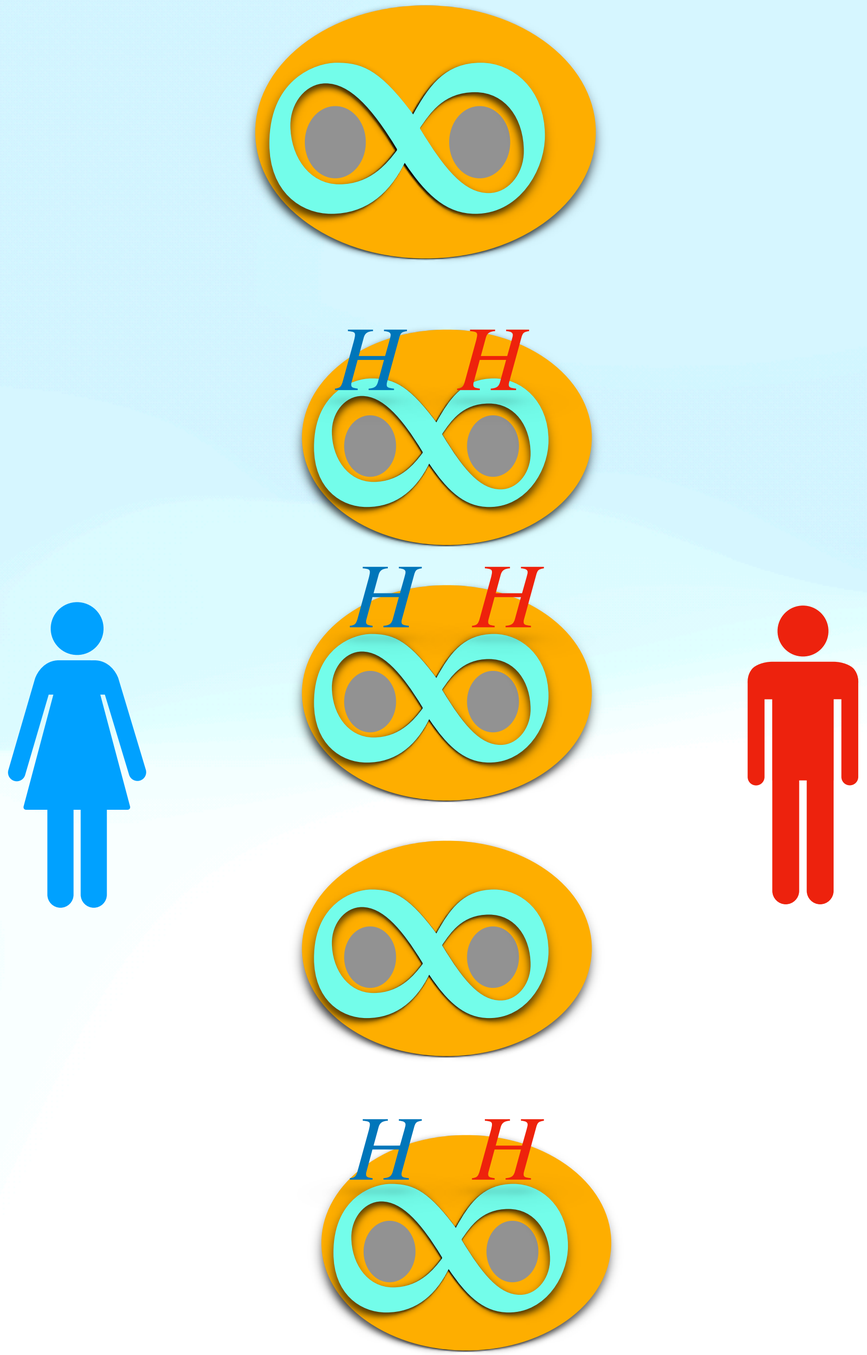


$\{b\}$ 0 1 1 0 ⋯ 1



Position of check bits




H H
H H
H H

Entanglement-based version of BB84 protocol

1.Alice creates $2n$ qubit pairs in the state $|\Phi^+\rangle^{\otimes 2n}$ with $|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$.

Entanglement-based version of BB84 protocol

1.Alice: creates $2n$ qubit pairs in the state $|\Phi^+\rangle^{\otimes 2n}$ with $|\Phi^+\rangle = \frac{1}{\sqrt{2}}\Big(|00\rangle + |11\rangle\Big)$.

2. Randomly selects n of these qubits that will later be used to estimate the errors in the qubit pairs.

Entanglement-based version of BB84 protocol

1. Alice creates $2n$ qubit pairs in the state $|\Phi^+\rangle^{\otimes 2n}$ with $|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$.

2. She randomly selects n of these qubits that will later be used to estimate the errors in the qubit pairs.

3. Selects a random classical bit string $b = (b_1, b_2, \cdots, b_{2n})$ of length $2n$. When $b_i = 1$, she applies a Hadamard transform to her half of the corresponding qubit pair.

Entanglement-based version of BB84 protocol

1. Alice creates $2n$ qubit pairs in the state $|\Phi^+\rangle^{\otimes 2n}$ with $|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$.
2. She randomly selects n of these qubits that will later be used to estimate the errors in the qubit pairs.
3. Alice selects a random classical bit string $b = (b_1, b_2, \cdots, b_{2n})$ of length $2n$. When $b_i = 1$, she applies a Hadamard transform to her half of the corresponding qubit pair.
4. She sends the other half of all qubit pairs to Bob.

Entanglement-based version of BB84 protocol

1. Alice creates $2n$ qubit pairs in the state $|\Phi^+\rangle^{\otimes 2n}$ with $|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$.
2. She randomly selects n of these qubits that will later be used to estimate the errors in the qubit pairs.
3. Alice selects a random classical bit string $b = (b_1, b_2, \cdots, b_{2n})$ of length $2n$. When $b_i = 1$, she applies a Hadamard transform to her half of the corresponding qubit pair.
4. She sends the other half of all qubit pairs to Bob.
5. Alice announces the string b and the positions of the check qubits.

Entanglement-based version of BB84 protocol

1. Alice creates $2n$ qubit pairs in the state $|\Phi^+\rangle^{\otimes 2n}$ with $|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$.
2. She randomly selects n of these qubits that will later be used to estimate the errors in the qubit pairs.
3. Alice selects a random classical bit string $b = (b_1, b_2, \cdots, b_{2n})$ of length $2n$. When $b_i = 1$, she applies a Hadamard transform to her half of the corresponding qubit pair.
4. She sends the other half of all qubit pairs to Bob.
5. Alice announces the string b and the positions of the check qubits.
6. Bob applies a Hadamard transformation to those qubits for which $b_i = 1$.

Entanglement-based version of BB84 protocol

1. Alice creates $2n$ qubit pairs in the state $|\Phi^+\rangle^{\otimes 2n}$ with $|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$.
2. She randomly selects n of these qubits that will later be used to estimate the errors in the qubit pairs.
3. Alice selects a random classical bit string $b = (b_1, b_2, \cdots, b_{2n})$ of length $2n$. When $b_i = 1$, she applies a Hadamard transform to her half of the corresponding qubit pair.
4. She sends the other half of all qubit pairs to Bob.
5. Alice announces the string b and the positions of the check qubits.
6. Bob applies a Hadamard transformation to those qubits for which $b_i = 1$.
7. Alice and Bob measure the check qubits in the computational basis $\{|0\rangle, |1\rangle\}$ to estimate the error rate. If more than t errors occur, they abort the protocol.

Entanglement-based version of BB84 protocol

1. Alice creates $2n$ qubit pairs in the state $|\Phi^+\rangle^{\otimes 2n}$ with $|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$.
2. She randomly selects n of these qubits that will later be used to estimate the errors in the qubit pairs.
3. Alice selects a random classical bit string $b = (b_1, b_2, \cdots, b_{2n})$ of length $2n$. When $b_i = 1$, she applies a Hadamard transform to her half of the corresponding qubit pair.
4. She sends the other half of all qubit pairs to Bob.
5. Alice announces the string b and the positions of the check qubits.
6. Bob applies a Hadamard transformation to those qubits for which $b_i = 1$.
7. Alice and Bob measure the check qubits in the computational basis $\{|0\rangle, |1\rangle\}$ to estimate the error rate. If more than t errors occur, they abort the protocol.
8. If the number of errors is below than t, Alice and Bob use the error correction codes C_1 and C_2 to correct the errors in the n remaining bits and obtain $|\Phi^+\rangle^{\otimes m}$.

Entanglement-based version of BB84 protocol

1. Alice creates $2n$ qubit pairs in the state $|\Phi^+\rangle^{\otimes 2n}$ with $|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$.
2. She randomly selects n of these qubits that will later be used to estimate the errors in the qubit pairs.
3. Alice selects a random classical bit string $b = (b_1, b_2, \cdots, b_{2n})$ of length $2n$. When $b_i = 1$, she applies a Hadamard transform to her half of the corresponding qubit pair.
4. She sends the other half of all qubit pairs to Bob.
5. Alice announces the string b and the positions of the check qubits.
6. Bob applies a Hadamard transformation to those qubits for which $b_i = 1$.
7. Alice and Bob measure the check qubits in the computational basis $\{|0\rangle, |1\rangle\}$ to estimate the error rate. If more than t errors occur, they abort the protocol.
8. If the number of errors is below than t, Alice and Bob use the error correction codes C_1 and C_2 to correct the errors in the n remaining bits and obtain $|\Phi^+\rangle^{\otimes m}$.
9. They measure the state $|\Phi^+\rangle^{\otimes m}$ in the computational basis to obtain the shred secret key.

Lemma

Let $\epsilon \geq 0$ and ρ_{AB} be a bipartite quantum system such that

$$F\left(\rho_{AB}, |\phi^+\rangle^{\otimes m}\right) \geq 1 - \epsilon^2.$$

Then the two $n-$bit strings resulting from local measurements of ρ_{AB} in the computational basis are $\epsilon-$secure keys (with respect to an adversary who controls a purifying system of ρ_{AB}.)

Ulhmann theorem

Relation between trace distance and fidelity

Lemma

Let $\epsilon \geq 0$ and ρ_{AB} be a bipartite quantum system such that

$$F\left(\rho_{AB}, |\phi^{+}\rangle^{\otimes m}\right) \geq 1 - \epsilon^{2}.$$

Then the two $n-$bit strings resulting from local measurements of ρ_{AB} in the computational basis are $\epsilon-$secure keys (with respect to an adversary who controls a purifying system of ρ_{AB}.)

Purifying system

$|\psi_{ABE}\rangle$ Such that $\rho_{AB} = Tr_{E}\left(|\psi_{ABE}\rangle\langle\psi_{ABE}|\right)$

$$\rho_{E} = Tr_{AB}\left(|\psi_{ABE}\rangle\langle\psi_{ABE}|\right)$$

Criterion for a protocol to be secure:

$$\frac{1}{2}\left|\left|\rho_{ABE}-\rho_{UU}\otimes\rho_{E}\right|\right|_{1}\leq\epsilon,$$

$$\rho_{UU}=\sum_{u\in\mathcal{S}}\frac{1}{|\mathcal{S}|}|u\rangle\langle u|\otimes|u\rangle\langle u|$$

$$F\left(\rho_{ABE},|\Phi^{+}\rangle^{\otimes m}\otimes\sigma_{E}\right)=F\left(\rho_{AB},|\phi^{+}\rangle^{\otimes m}\right)$$

$$\begin{aligned}\frac{1}{2}\left|\left|\rho_{ABE}-||\Phi^{+}\rangle^{\otimes m}\otimes\sigma_{E}\right|\right|_{1}&\leq\sqrt{1-F\left(\rho_{ABE},|\Phi^{+}\rangle^{\otimes m}\otimes\sigma_{E}\right)}\\&=\sqrt{1-F\left(\rho_{AB},|\Phi^{+}\rangle^{\otimes m}\right)}\\&\leq\sqrt{1-(1-\epsilon^{2})}=\epsilon\end{aligned}$$

Relation between trace distance and fidelity

Quantum measurements do allow an interpretation in terms of classical probability theory if the observables that are considered refer to only one basis.

$$|\Phi^{\pm}\rangle = \frac{1}{\sqrt{2}}\Big(|00\rangle \pm |11\rangle\Big), \quad |\Psi^{\pm}\rangle = \frac{1}{\sqrt{2}}\Big(|01\rangle \pm |10\rangle\Big)$$

In the protocol, the three errors are generated by $\sigma_x, \sigma_y, \sigma_z$.

Nothing $(\mathbf{1})$	$\|\Phi^+\rangle \rightarrow \|\Phi^+\rangle$
Bit flip (σ_x)	$\|\Phi^+\rangle \rightarrow \|\Psi^+\rangle$
Phase flip (σ_z)	$\|\Phi^+\rangle \rightarrow \|\Phi^-\rangle$
Bit+phase (σ_y)	$\|\Phi^+\rangle \rightarrow \|\Psi^-\rangle$

Detectors of errors: POVM

Detector of bit-flip

$$\Pi_{bf} = |\Psi^+\rangle\langle\Psi^+| + |\Psi^-\rangle\langle\Psi^-|$$

Detector of phase-error

$$\Pi_{pe} = |\Phi^-\rangle\langle\Phi^-| + |\Psi^-\rangle\langle\Psi^-|$$

$$\Pi_{bf} = \frac{1}{2}(\mathbf{1} \otimes \mathbf{1} - \sigma_z \otimes \sigma_z)$$

$$\Pi_{pe} = \frac{1}{2}(\mathbf{1} \otimes \mathbf{1} - \sigma_x \otimes \sigma_x)$$

Reduction to the prepare-and-measure version

The measurement of n check pairs in step 8 is simply used to estimate the error rate.

Instead of using entangled states, Alice can equivalently prepare and send n single qubit states to Bob. This changes sets 1, 2 and 8 of the protocol to:

Entanglement-based version of BB84 protocol

1. Alice creates n random check bits and n qubit pairs in the state $|\Phi^+\rangle^{\otimes n}$. She encodes n qubits as $|0\rangle$ or $|1\rangle$ according to the check bits.
2. She randomly chooses n out of $2n$ positions to put in the check qubits. In the remaining positions, she puts in one half of each state $|\Phi^+\rangle$.
3. Alice selects a random classical bit string $b = (b_1, b_2, \cdots, b_{2n})$ of length $2n$. When $b_i = 1$, she applies a Hadamard transform to her half of the corresponding qubit pair.
4. She sends the other half of all qubit pairs to Bob.
5. Alice announces the string b and the positions of the check qubits.
6. Bob applies a Hadamard transformation to those qubits for which $b_i = 1$.
7. Alice and Bob measure the check qubits in the computational basis $\{|0\rangle, |1\rangle\}$ to estimate the error rate. If more than t errors occur, they abort the protocol.
8. Bob measures the n check qubits in the $\{|0\rangle, |1\rangle\}$ basis and publicly shares the result with Alice. If more than t bits disagree, they abort the protocol.
9. They measure the state $|\Phi^+\rangle^{\otimes m}$ in the computational basis to obtain the shred secret key.

How to remove the remaining n entangled pairs?

Two classical linear error correction codes C_1 (encoding k_1 bits into n bits) and C_2 (encoding k_2 bits into n bits), a $[n, m]$ CSS code of C_1 over C_2 encodes $m = k_1 - k_2$ qubits into n qubits and corrects upto t errors.

Position of bit-flip error: parity check matrix H_1 of the classical code C_1

Information about the phase-error: parity check matrix $H_2^{\perp}$ of the classical dual code $C_2^{\perp}$

A codeword in this code is always of the form

$$|x_k + C_2\rangle = \frac{1}{\sqrt{|C_2|}} \sum_{y \in C_2} |x_k + y\rangle,$$

x_k: a representative of one of the 2^m cosets of C_2 in C_1.

x_k : a vector x indexed by a string k .

$$|x_k, v, w\rangle = \frac{1}{\sqrt{|C_2|}} \sum_{y \in C_2} (-1)^{v \cdot y} |x_k + y + w\rangle$$

x_k : a vector x indexed by a string k .

$$|x_k, v, w\rangle = \frac{1}{\sqrt{|C_2|}} \sum_{y \in C_2} (-1)^{v \cdot y} |x_k + y + w\rangle$$

$[n, m]$ CSS Codes of C_1 over C_2.

These states form an orthonormal basis of a 2^n dimensional Hilbert space.

x_k : a vector x indexed by a string k .

$$|x_k, v, w\rangle = \frac{1}{\sqrt{|C_2|}} \sum_{y \in C_2} (-1)^{v \cdot y} |x_k + y + w\rangle$$

$[n, m]$ CSS Codes of C_1 over C_2.

These states form an orthonormal basis of a 2^n dimensional Hilbert space.

We show that there are $2^{k_1 - k_2}$ distinct values of x_k, 2^{n-k_1} distinct values of w and 2^{k_2} distinct values of v .

x_k : a vector x indexed by a string k .

$$|x_k, v, w\rangle = \frac{1}{\sqrt{|C_2|}} \sum_{y \in C_2} (-1)^{v \cdot y} |x_k + y + w\rangle$$

$[n, m]$ CSS Codes of C_1 over C_2.

These states form an orthonormal basis of a 2^n dimensional Hilbert space.

We show that there are $2^{k_1 - k_2}$ distinct values of x_k, 2^{n-k_1} distinct values of w and 2^{k_2} distinct values of v .

I. If $x_k - x_k' \in C_2$, then $|x_k + C_2\rangle = |x_k' + C_2\rangle$, which implies that the state $|x_k + C_2\rangle$ only depends on the coset C_1/C_2 in which x_k is contained. Since there are 2^m such cosets, there are 2^m distinct values of x_k .

x_k : a vector x indexed by a string k .

$$|x_k, v, w\rangle = \frac{1}{\sqrt{|C_2|}} \sum_{y \in C_2} (-1)^{v \cdot y} |x_k + y + w\rangle$$

$[n, m]$ CSS Codes of C_1 over C_2.

These states form an orthonormal basis of a 2^n dimensional Hilbert space.

We show that there are $2^{k_1 - k_2}$ distinct values of x_k, 2^{n-k_1} distinct values of w and 2^{k_2} distinct values of v .

I. If $x_k - x_k' \in C_2$, then $|x_k + C_2\rangle = |x_k' + C_2\rangle$, which implies that the state $|x_k + C_2\rangle$ only depends on the coset C_1/C_2 in which x_k is contained. Since there are 2^m such cosets, there are 2^m distinct values of x_k .

2. Suppose that $|x_k, v, w\rangle = |x_k, v', w\rangle$. This implies that $v \cdot y = v' \cdot Y$ for all $y \in C_2$, and therefore $(v - v') \cdot y = 0$ for all $y \in C_2$. This means that $v - v' \in K$, where K is the row space of the parity check matrix H_2 of C_2. The rows of H_2 span the space of all the vectors that are orthogonal to the codewords y of C_2.

Therefore, the requirement for two states $|x_k, v, w\rangle$ and $|x_k, v', w\rangle$ to be distinct states is $v - v' \notin K$, which directly implies $v + K \neq v' + K$ (which is a property of cosets). Since H_2 has $n - k_2$ linearly independent rows, v has $2^{n-(n-k_2)} = 2^{k_2}$ distinct values.

III. With a similar argument as in I, the values of w depend on the coset of C_1 in a 2^n dimensional Hilbert space in which w is contained, which implies that there are 2^{n-k_1} distinct values of w .

Orthonormality of the states

If x_k and x_k' belong to different cosets of C_1/C_2, then there are no codewords $y, y' \in C_2$ such that $x_k + y = x_k' + y'$. Hence, the states are orthonormal. A similar argument can be given in the current situation.

Consider the cosets of $C_2 \in F_2^n$ (the set of all $n-$ bit strings. For two distinct states $|x_k, v, w\rangle$ and $|x_k', v', w'\rangle, x_k + w$ and $x_k' + w$ belong to different cosets of F_2^n/C_2, and therefore, there are no $y, y' \in C_2$ such that $x_k + w + y = x_k' + w' + y'$; hence the two states are orthonormal.)