Procesamiento Cuántico de Información

Ariel Bendersky[1]

[1]Departamento de Computación - FCEyN - Universidad de Buenos Aires

Clase 5

Clase 5

- Algoritmo de Deutsch.
- Algoritmo de Deutsch-Jozsa.
- Criptografía cuántica.

El algoritmo de Deutsch

El problema clásico

Sea una función $f : \{0, 1\} \to f : \{0, 1\}$ desconocida. Se quiere saber si f es constante ($f(0) = f(1)$) o no. ¿Cuántas veces es necesario evaluar f para saber eso?

Solución clásica

Se ve trivialmente que hay que averiguar tanto $f(0)$ como $f(1)$ y, por lo tanto, es necesario evaluar la función dos veces para determinar si la función es constante.

Ventaja cuántica

Evaluando la función f una sola vez podemos determinar si es constante o no.

El algoritmo de Deutsch

El problema clásico

Sea una función $f : \{0,1\} \to f : \{0,1\}$ desconocida. Se quiere saber si f es constante ($f(0) = f(1)$) o no. ¿Cuántas veces es necesario evaluar f para saber eso?

Solución clásica

Se ve trivialmente que hay que averiguar tanto $f(0)$ como $f(1)$ y, por lo tanto, es necesario evaluar la función dos veces para determinar si la función es constante.

Ventaja cuántica

Evaluando la función f una sola vez podemos determinar si es constante o no.

El algoritmo de Deutsch

El problema clásico

Sea una función $f : \{0, 1\} \to f : \{0, 1\}$ desconocida. Se quiere saber si f es constante $(f(0) = f(1))$ o no. ¿Cuántas veces es necesario evaluar f para saber eso?

Solución clásica

Se ve trivialmente que hay que averiguar tanto $f(0)$ como $f(1)$ y, por lo tanto, es necesario evaluar la función dos veces para determinar si la función es constante.

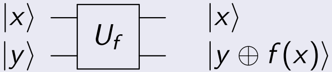
Ventaja cuántica

Evaluando la función f una sola vez podemos determinar si es constante o no.

El algoritmo de Deutsch

Evaluando f

Nos dan una unitaria U_f que codifica f y actúa sobre dos qubits como:



donde $\oplus$ es la suma módulo 2.

Nota

Eso sirve también para evaluar la función clásicamente.

El algoritmo de Deutsch

Evaluando f

Nos dan una unitaria U_f que codifica f y actúa sobre dos qubits como:

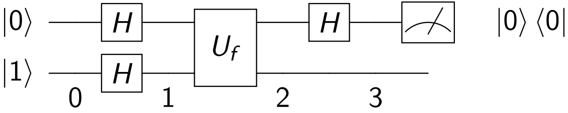
$$\begin{array}{lcl} |x\rangle & \multirow{2}{*}{$\boxed{U_f}$} & |x\rangle \\ |y\rangle & & |y \oplus f(x)\rangle \end{array}$$

donde $\oplus$ es la suma módulo 2.

Nota

Eso sirve también para evaluar la función clásicamente.

El algoritmo de Deutsch



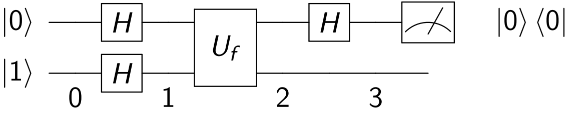
Análisis

$$|\psi_0\rangle = |0\rangle|1\rangle$$

$$|\psi_1\rangle = \frac{1}{2}(|0\rangle + |1\rangle)(|0\rangle - |1\rangle) = \frac{1}{2}(|00\rangle - |01\rangle + |10\rangle - |11\rangle)$$

$$|\psi_2\rangle = \frac{1}{2}(|0\rangle|f(0)\rangle - |0\rangle|\neg f(0)\rangle + |1\rangle|f(1)\rangle - |1\rangle|\neg f(1)\rangle)$$

El algoritmo de Deutsch



Análisis

$$|\psi_0\rangle = |0\rangle |1\rangle$$

$$|\psi_1\rangle = \frac{1}{2}\left(|0\rangle + |1\rangle\right)\left(|0\rangle - |1\rangle\right) = \frac{1}{2}\left(|00\rangle - |01\rangle + |10\rangle - |11\rangle\right)$$

$$|\psi_2\rangle = \frac{1}{2}\left(|0\rangle |f(0)\rangle - |0\rangle |\neg f(0)\rangle + |1\rangle |f(1)\rangle - |1\rangle |\neg f(1)\rangle\right)$$

El algoritmo de Deutsch

Análisis

$$|\psi_2\rangle = \frac{1}{2}\left(|0\rangle\,|f(0)\rangle - |0\rangle\,|\neg f(0)\rangle + |1\rangle\,|f(1)\rangle - |1\rangle\,|\neg f(1)\rangle\right)$$

Si $f(0) = f(1) = c$:

$$\begin{aligned}|\psi_2^{=}\rangle = \frac{1}{2}\left(\left(|0\rangle + |1\rangle\right)|c\rangle - \left(|0\rangle + |1\rangle\right)|\neg c\rangle\right) = \\ = \frac{1}{2}\left(|0\rangle + |1\rangle\right)\left(|c\rangle - |\neg c\rangle\right)\end{aligned}$$

Si $f(0) = c = \neg f(1)$:

$$\begin{aligned}|\psi_2^{\neq}\rangle = \frac{1}{2}\left(|0\rangle\,|c\rangle - |0\rangle\,|\neg c\rangle + |1\rangle\,|\neg c\rangle - |1\rangle\,|c\rangle\right) = \\ = \frac{1}{2}\left(|0\rangle - |1\rangle\right)\left(|c\rangle - |\neg c\rangle\right)\end{aligned}$$

El algoritmo de Deutsch

Análisis

$$|\psi_3^{=}\rangle = |0\rangle \frac{1}{\sqrt{2}} (|c\rangle - |\neg c\rangle)$$

$$|\psi_3^{\neq}\rangle = |1\rangle \frac{1}{\sqrt{2}} (|c\rangle - |\neg c\rangle)$$

Midiendo el primer registro determino si la función es constante o no.

El algoritmo de Deutsch

Implementación.

El algoritmo de Deutsch-Jozsa

Definición

Es una extensión del algoritmo de Deutsch.
Dada $f : \{0,1\}^n \to \{0,1\}$ y la promesa de que f es constante o balanceada (i.e., tiene la misma cantidad de entradas para las que la salida da 1 que para las que da 0), se desea saber si es constante o balanceada.

Clásicamente

El mejor algoritmo clásico necesita evaluar f en $2^{n-1}+1$ entradas para dar la respuesta con seguridad.

Cuánticamente

El algoritmo cuántico sólo evalúa la función f una vez y da la respuesta correcta.

El algoritmo de Deutsch-Jozsa

Definición

Es una extensión del algoritmo de Deutsch.
Dada $f : \{0,1\}^n \to \{0,1\}$ y la promesa de que f es constante o balanceada (i.e., tiene la misma cantidad de entradas para las que la salida da 1 que para las que da 0), se desea saber si es constante o balanceada.

Clásicamente

El mejor algoritmo clásico necesita evaluar f en $2^{n-1}+1$ entradas para dar la respuesta con seguridad.

Cuánticamente

El algoritmo cuántico sólo evalúa la función f una vez y da la respuesta correcta.

El algoritmo de Deutsch-Jozsa

Definición

Es una extensión del algoritmo de Deutsch.
Dada $f : \{0,1\}^n \to \{0,1\}$ y la promesa de que f es constante o balanceada (i.e., tiene la misma cantidad de entradas para las que la salida da 1 que para las que da 0), se desea saber si es constante o balanceada.

Clásicamente

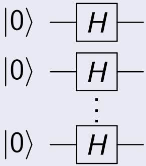
El mejor algoritmo clásico necesita evaluar f en $2^{n-1}+1$ entradas para dar la respuesta con seguridad.

Cuánticamente

El algoritmo cuántico sólo evalúa la función f una vez y da la respuesta correcta.

El algoritmo de Deutsch-Jozsa

Resultado auxiliar



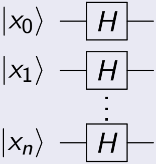
Tenemos

$$H^{\otimes n} |0\rangle^{\otimes n} = \frac{1}{\sqrt{2^n}} (|0\rangle + |1\rangle) \cdots (|0\rangle + |1\rangle)$$

$$= \frac{1}{\sqrt{2^n}} \sum_{z=0}^{2^n - 1} |Z\rangle$$

El algoritmo de Deutsch-Jozsa

En general



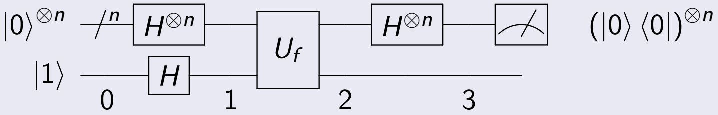
Tenemos

$$H^{\otimes n}|x\rangle = \sum_{z=0}^{2^n-1} (-1)^{x \cdot z} |z\rangle$$

donde $x \cdot z$ es el producto interno bit a bit.

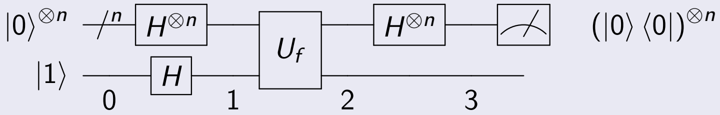
El algoritmo de Deutsch-Jozsa

El circuito



El algoritmo de Deutsch-Jozsa

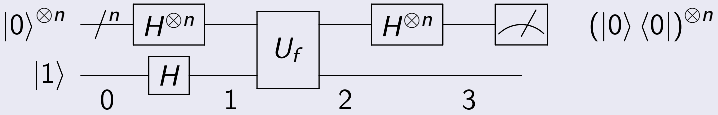
El circuito



$$|\psi_0\rangle = |0\rangle^{\otimes n}|1\rangle$$

El algoritmo de Deutsch-Jozsa

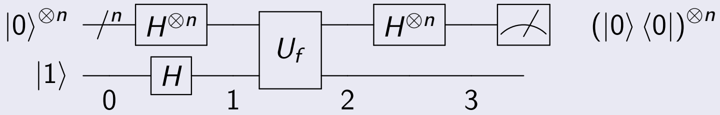
El circuito



$$|\psi_1\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right)$$

El algoritmo de Deutsch-Jozsa

El circuito

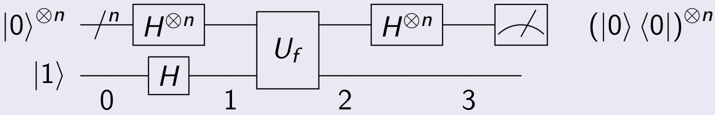


$$|\psi_2\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle \left(\frac{|0 \oplus f(x)\rangle - |1 \oplus f(x)\rangle}{\sqrt{2}} \right)$$

$f(x)$ sólo saca un factor -1 si $f(x) = 1$ y 0 en otro caso.

El algoritmo de Deutsch-Jozsa

El circuito

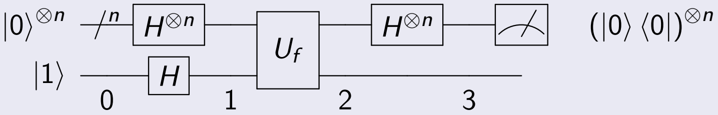


Reescribimos como

$$|\psi_2\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} (-1)^{f(x)} |x\rangle \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right)$$

El algoritmo de Deutsch-Jozsa

El circuito

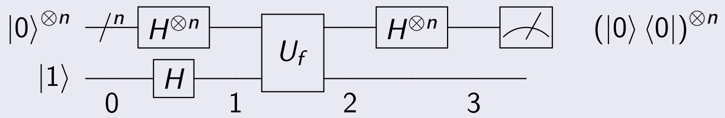


Aplicamos $H^{\otimes n}$:

$$|\psi_3\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} (-1)^{f(x)} \sum_{z=0}^{2^n-1} (-1)^{x \cdot z} |z\rangle \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right)$$

El algoritmo de Deutsch-Jozsa

El circuito



Reescribimos:

$$|\psi_3\rangle = \frac{1}{\sqrt{2^n}} \sum_{z=0}^{2^n-1} \sum_{x=0}^{2^n-1} (-1)^{f(x)+x\cdot z} |z\rangle \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right)$$

Si f es constante, sólo sobrevive el $z = 0$. Si no es constante, $z = 0$ desaparece.

El algoritmo de Deutsch-Jozsa

- El algoritmo cuántico tiene una ventaja exponencial.
- No es muy claro que evaluar f de esa forma de una comparación justa con el caso clásico.
- Además, si comparamos con el algoritmo clásico probabilístico, la ventaja desaparece.

El algoritmo de Deutsch-Jozsa

- El algoritmo cuántico tiene una ventaja exponencial.
- No es muy claro que evaluar f de esa forma de una comparación justa con el caso clásico.
- Además, si comparamos con el algoritmo clásico probabilístico, la ventaja desaparece.

El algoritmo de Deutsch-Jozsa

- El algoritmo cuántico tiene una ventaja exponencial.
- No es muy claro que evaluar f de esa forma de una comparación justa con el caso clásico.
- Además, si comparamos con el algoritmo clásico probabilístico, la ventaja desaparece.

Distribución cuántica de claves

Criptografía cuántica

One time pad

Los *one time pad* o libretas de un solo uso, proveen criptografía segura, siempre y cuando se utilice la clave una única vez (es decir, clave tan larga como el mensaje), y sea secreta de Alice y Bob.

- Alice y Bob comparten un string binario (el pad, clave o libreta).
- Alice encripta su mensaje usando XOR.
- Alice le manda el el mensaje a Bob por un canal público.
- Bob desencripta usando XOR.

One time pad

Clave compartida : 100110011110

Clave	1	0	0	1	1	0	0	1	1	1	1	0
Mensaje	0	1	0	1	1	1	0	1	0	0	0	1
XOR	1	1	0	0	0	1	0	0	1	1	1	1

Alice envía el XOR (para un espía eso es ruido). Bob recibe y hace lo mismo.

Clave	1	0	0	1	1	0	0	1	1	1	1	0
Mensaje enc.	1	1	0	0	0	1	0	0	1	1	1	1
XOR	0	1	0	1	1	1	0	1	0	0	0	1

Alice recupera el mensaje de Bob.

One time pad

Clave compartida : 100110011110

Clave	1	0	0	1	1	0	0	1	1	1	1	0
Mensaje	0	1	0	1	1	1	0	1	0	0	0	1
XOR	1	1	0	0	0	1	0	0	1	1	1	1

Alice envía el XOR (para un espía eso es ruido). Bob recibe y hace lo mismo.

Clave	1	0	0	1	1	0	0	1	1	1	1	0
Mensaje enc.	1	1	0	0	0	1	0	0	1	1	1	1
XOR	0	1	0	1	1	1	0	1	0	0	0	1

Alice recupera el mensaje de Bob.

One time pad

Clave compartida : 100110011110

Clave	1	0	0	1	1	0	0	1	1	1	1	0
Mensaje	0	1	0	1	1	1	0	1	0	0	0	1
XOR	1	1	0	0	0	1	0	0	1	1	1	1

Alice envía el XOR (para un espía eso es ruido). Bob recibe y hace lo mismo.

Clave	1	0	0	1	1	0	0	1	1	1	1	0
Mensaje enc.	1	1	0	0	0	1	0	0	1	1	1	1
XOR	0	1	0	1	1	1	0	1	0	0	0	1

Alice recupera el mensaje de Bob.

One time pad

Clave compartida : 100110011110

Clave	1	0	0	1	1	0	0	1	1	1	1	0
Mensaje	0	1	0	1	1	1	0	1	0	0	0	1
XOR	1	1	0	0	0	1	0	0	1	1	1	1

Alice envía el XOR (para un espía eso es ruido). Bob recibe y hace lo mismo.

Clave	1	0	0	1	1	0	0	1	1	1	1	0
Mensaje enc.	1	1	0	0	0	1	0	0	1	1	1	1
XOR	0	1	0	1	1	1	0	1	0	0	0	1

Alice recupera el mensaje de Bob.

Distribución cuántica de claves

La cuántica nos da un método seguro de generar claves secretas para Alice y Bob.
Primeros protocolos:

- Benentt y Brassard 1984 (Preparar y medir).
- Ekert 1991 (Basado en entrelazamiento).
- BB92 (Preparar y medir).

BB92 - El protocolo sin ruido ni espías

1. Alice prepara al azar uno de los estados $|0\rangle$ o $\frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right)$ y se lo manda a Bob.
2. Bob le mide a ese estado al azar en alguna de las bases $B_1 = \{|0\rangle, |1\rangle\}$ o $B_2 = \left\{\frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right), \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right)\right\}$.
3. Cuando obtiene el segundo estado de la base en que midió, sabe con certeza el estado que preparó Alice. En ese caso, anuncia públicamente que tuvo éxito. Alice y Bob ya tienen un primer bit de clave secreta.
4. Vuelven al paso 1 las veces que sea necesario.

BB92 - Detectando espías

Si en el camino por el cual Alice le manda a Bob un estado, hubo un espía midiendo al estado que viajaba y reenvíando después de medir, cada vez que el espía no adivinaba la base en la que midió Alice, estaría reenvíando un estado distinto al que mandó Alice. ¿Cómo hacen Alice y Bob para detectarlo?

- Por un canal público, Alice y Bob se ponen de acuerdo y publican algunos bits de sus claves.
- Si los bits publicados son iguales, se quedan tranquilos.
- Si son distintos, saben que algo pasó en el camino.
- Si publican n bits de clave y había un espía, la probabilidad de detectarlo es $p_d = 1 - (3/4)^n$

Otros tipos de ataque

- Man in the middle.
- División del número de fotones.
- Trojan horse (iluminar el canal cuántico y ver las reflexiones para saber en qué base midió Bob).

BB84 - El protocolo (Simplificado)

Bases: $B_Z = \{|0\rangle, |1\rangle\}$ (base computacional, $+$), $B_X = \{|+\rangle, |-\rangle\}$ (base Hadamard/diagonal, X).

1. **Transmisión Cuántica:**
   - Alice elige al azar una secuencia de bits k_A y una secuencia de bases b_A.
   - Para cada bit i, prepara un qubit en el estado correspondiente al bit $k_{A,i}$ usando la base $b_{A,i}$.
   - Envía los qubits a Bob por el canal cuántico.
2. **Medición:**
   - Bob elige al azar una secuencia de bases de medición b_B.
   - Mide cada qubit recibido en la base $b_{B,i}$ y registra el resultado $k_{B,i}$.

BB84 - El protocolo (Simplificado)

- **Sifting (Cernido):**
  - Alice y Bob *publican* (por canal clásico) sus secuencias de bases (b_A, b_B).
  - Se quedan *sólo* con los bits $k_{A,i}, k_{B,i}$ donde usaron la misma base $(b_{A,i} = b_{B,i})$.
  - Esta es la *clave cruda* (raw key). Idealmente $k_A = k_B$ en esta clave.

Detectando a Eve

¿Cómo saben Alice y Bob si Eve estuvo escuchando?

- **Principio clave:** Si Eve intercepta un qubit, lo mide y lo reenvía, inevitablemente introducirá errores si elige la base incorrecta.
- **Procedimiento:**
  1. Alice y Bob acuerdan (públicamente) sacrificar una *fracción aleatoria* de los bits de su clave cruda (los obtenidos después del sifting).
  2. Comparan (públicamente) los valores de estos bits sacrificados.
  3. Calculan la Tasa de Error Cuántico de Bits (QBER - Quantum Bit Error Rate):

$$\text{QBER} = \frac{\text{Número de bits diferentes}}{\text{Número total de bits comparados}}$$

Detectando a Eve

¿Cómo saben Alice y Bob si Eve estuvo escuchando?

- **Principio clave:** Si Eve intercepta un qubit, lo mide y lo reenvía, inevitablemente introducirá errores si elige la base incorrecta.
- **Procedimiento:**
  1. Alice y Bob acuerdan (públicamente) sacrificar una *fracción aleatoria* de los bits de su clave cruda (los obtenidos después del sifting).
  2. Comparan (públicamente) los valores de estos bits sacrificados.
  3. Calculan la Tasa de Error Cuántico de Bits (QBER - Quantum Bit Error Rate):

$$\text{QBER} = \frac{\text{Número de bits diferentes}}{\text{Número total de bits comparados}}$$

Detectando a Eve

- **Decisión:**
  - Si QBER es *bajo* (por debajo de un umbral predefinido ϵ), asumen que la interferencia (ruido o Eve) es tolerable y continúan.
  - Si QBER es *alto*, asumen la presencia de Eve y abortan el protocolo. ¡La clave no es segura!

¿Por qué Eve introduce errores? (BB84 - Parte 1)

Supongamos que Alice envía $|0\rangle$ (bit 0, base Z). Bob (si no hay Eve) mediría en base Z y obtendría $|0\rangle$.

- **Eve intercepta:** No sabe la base de Alice (Z). Mide al azar en base Z o X.
- **Caso 1: Eve elige base Z (50% prob.)**
  - Eve mide $|0\rangle$.
  - Reenvía $|0\rangle$ a Bob.
  - Bob (que eligió base Z) mide $|0\rangle$.
  - *Resultado:* No hay error. Eve no es detectada (en este bit).
- **Caso 2: Eve elige base X (50% prob.)**
  - Medir $|0\rangle$ en base X ($= \{|+\rangle, |-\rangle\}$) da $|+\rangle$ o $|-\rangle$ (con 50% prob. c/u).
  - Veremos qué pasa en el siguiente slide...

¿Por qué Eve introduce errores? (BB84 - Parte 1)

Supongamos que Alice envía $|0\rangle$ (bit 0, base Z). Bob (si no hay Eve) mediría en base Z y obtendría $|0\rangle$.

- **Eve intercepta:** No sabe la base de Alice (Z). Mide al azar en base Z o X.
- **Caso 1: Eve elige base Z (50% prob.)**
  - Eve mide $|0\rangle$.
  - Reenvía $|0\rangle$ a Bob.
  - Bob (que eligió base Z) mide $|0\rangle$.
  - *Resultado:* No hay error. Eve no es detectada (en este bit).
- **Caso 2: Eve elige base X (50% prob.)**
  - Medir $|0\rangle$ en base X ($= \{|+\rangle, |-\rangle\}$) da $|+\rangle$ o $|-\rangle$ (con 50% prob. c/u).
  - Veremos qué pasa en el siguiente slide...

¿Por qué Eve introduce errores? (BB84 - Parte 1)

Supongamos que Alice envía $|0\rangle$ (bit 0, base Z). Bob (si no hay Eve) mediría en base Z y obtendría $|0\rangle$.

- **Eve intercepta:** No sabe la base de Alice (Z). Mide al azar en base Z o X.
- **Caso 1: Eve elige base Z (50% prob.)**
  - Eve mide $|0\rangle$.
  - Reenvía $|0\rangle$ a Bob.
  - Bob (que eligió base Z) mide $|0\rangle$.
  - *Resultado:* No hay error. Eve no es detectada (en este bit).
- **Caso 2: Eve elige base X (50% prob.)**
  - Medir $|0\rangle$ en base X ($= \{|+\rangle, |-\rangle\}$) da $|+\rangle$ o $|-\rangle$ (con 50% prob. c/u).
  - Veremos qué pasa en el siguiente slide...

¿Por qué Eve introduce errores? (BB84 - Parte 2)

Continuación Caso 2: Eve elige base X (50% prob.)

- Supongamos que Eve midió $|+\rangle$ y reenvía $|+\rangle$ a Bob.
- Bob (que eligió medir en base Z) recibe $|+\rangle$.
- Bob mide $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ en base Z.
  - Obtiene $|0\rangle$ (resultado correcto) con prob. 1/2.
  - Obtiene $|1\rangle$ (**ERROR**) con prob. 1/2.

Conclusión:

- Si Eve mide en la base incorrecta (prob. 1/2), introduce un error con probabilidad 1/2.
- **Probabilidad total de error inducido por Eve (por bit sifteado):**

$$P(\text{Error}) = P(\text{Eve elige X}) \times P(\text{Error}|\text{Eve midió X})$$
$$= \frac{1}{2} \times \frac{1}{2} = \frac{1}{4}$$

¡La intercepción de Eve introduce errores detectables!

¿Por qué Eve introduce errores? (BB84 - Parte 2)

Continuación Caso 2: Eve elige base X (50% prob.)

- Supongamos que Eve midió $|+\rangle$ y reenvía $|+\rangle$ a Bob.
- Bob (que eligió medir en base Z) recibe $|+\rangle$.
- Bob mide $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ en base Z.
  - Obtiene $|0\rangle$ (resultado correcto) con prob. $1/2$.
  - Obtiene $|1\rangle$ (**ERROR**) con prob. $1/2$.

Conclusión:

- Si Eve mide en la base incorrecta (prob. $1/2$), introduce un error con probabilidad $1/2$.
- **Probabilidad total de error inducido por Eve (por bit sifteado):**

$$P(\text{Error}) = P(\text{Eve elige X}) \times P(\text{Error}|\text{Eve midió X})$$
$$= \frac{1}{2} \times \frac{1}{2} = \frac{1}{4}$$

¡La intercepción de Eve introduce errores detectables!

¿Por qué Eve introduce errores? (BB84 - Parte 2)

Continuación Caso 2: Eve elige base X (50% prob.)

- Supongamos que Eve midió $|+\rangle$ y reenvía $|+\rangle$ a Bob.
- Bob (que eligió medir en base Z) recibe $|+\rangle$.
- Bob mide $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ en base Z.
  - Obtiene $|0\rangle$ (resultado correcto) con prob. $1/2$.
  - Obtiene $|1\rangle$ (**ERROR**) con prob. $1/2$.

Conclusión:

- Si Eve mide en la base incorrecta (prob. $1/2$), introduce un error con probabilidad $1/2$.
- **Probabilidad total de error inducido por Eve (por bit sifteado):**

$$P(\text{Error}) = P(\text{Eve elige X}) \times P(\text{Error}|\text{Eve midió X})$$
$$= \frac{1}{2} \times \frac{1}{2} = \frac{1}{4}$$

¡La intercepción de Eve introduce errores detectables!

¿Por qué Eve introduce errores? (BB84 - Parte 2)

Continuación Caso 2: Eve elige base X (50% prob.)

- Supongamos que Eve midió $|+\rangle$ y reenvía $|+\rangle$ a Bob.
- Bob (que eligió medir en base Z) recibe $|+\rangle$.
- Bob mide $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ en base Z.
  - Obtiene $|0\rangle$ (resultado correcto) con prob. $1/2$.
  - Obtiene $|1\rangle$ (**ERROR**) con prob. $1/2$.

Conclusión:

- Si Eve mide en la base incorrecta (prob. $1/2$), introduce un error con probabilidad $1/2$.
- **Probabilidad total de error inducido por Eve (por bit sifteado):**

$$P(\text{Error}) = P(\text{Eve elige X}) \times P(\text{Error}|\text{Eve midió X})$$
$$= \frac{1}{2} \times \frac{1}{2} = \frac{1}{4}$$

¡La intercepción de Eve introduce errores detectables!

¿Por qué Eve introduce errores? (BB84 - Parte 2)

Continuación Caso 2: Eve elige base X (50% prob.)

- Supongamos que Eve midió $|+\rangle$ y reenvía $|+\rangle$ a Bob.
- Bob (que eligió medir en base Z) recibe $|+\rangle$.
- Bob mide $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ en base Z.
  - Obtiene $|0\rangle$ (resultado correcto) con prob. 1/2.
  - Obtiene $|1\rangle$ (**ERROR**) con prob. 1/2.

Conclusión:

- Si Eve mide en la base incorrecta (prob. 1/2), introduce un error con probabilidad 1/2.
- **Probabilidad total de error inducido por Eve (por bit sifteado):**

$$P(\text{Error}) = P(\text{Eve elige X}) \times P(\text{Error}|\text{Eve midió X})$$

$$= \frac{1}{2} \times \frac{1}{2} = \frac{1}{4}$$

¡La intercepción de Eve introduce errores detectables!

Probabilidad de Detectar a Eve (Boceto - Parte 1)

Supongamos que Alice y Bob comparan n bits de su clave cruda (después del sifting). Asumimos el peor caso para A y B: Eve intercepta *todos* los qubits.

- **Para un bit comparado i:**
  - Sabemos que Eve introduce un error con probabilidad $1/4$.

$$P(\text{Error en bit } i) = 1/4$$

  - La probabilidad de que Eve NO introduzca error en ese bit es:

$$P(\text{No error en bit } i) = 1 - 1/4 = 3/4$$

- **Para los n bits comparados:**
  - Asumimos que los errores introducidos por Eve en cada bit son independientes.

Probabilidad de Detectar a Eve (Boceto - Parte 1)

Supongamos que Alice y Bob comparan n bits de su clave cruda (después del sifting). Asumimos el peor caso para A y B: Eve intercepta *todos* los qubits.

- **Para un bit comparado i:**
  - Sabemos que Eve introduce un error con probabilidad $1/4$.

$$P(\text{Error en bit } i) = 1/4$$

  - La probabilidad de que Eve NO introduzca error en ese bit es:

$$P(\text{No error en bit } i) = 1 - 1/4 = 3/4$$

- **Para los n bits comparados:**
  - Asumimos que los errores introducidos por Eve en cada bit son independientes.

Probabilidad de Detectar a Eve (Boceto - Parte 1)

Supongamos que Alice y Bob comparan n bits de su clave cruda (después del sifting). Asumimos el peor caso para A y B: Eve intercepta *todos* los qubits.

- **Para un bit comparado i:**
  - Sabemos que Eve introduce un error con probabilidad $1/4$.

$$P(\text{Error en bit } i) = 1/4$$

  - La probabilidad de que Eve NO introduzca error en ese bit es:

$$P(\text{No error en bit } i) = 1 - 1/4 = 3/4$$

- **Para los n bits comparados:**
  - Asumimos que los errores introducidos por Eve en cada bit son independientes.

Probabilidad de Detectar a Eve (Boceto - Parte 2)

Continuando con n bits comparados e independencia:

- Probabilidad de que Eve *no* introduzca error en *ninguno* de los n bits:

$$P(\text{Ningún error en } n \text{ bits}) = \left(\frac{3}{4}\right)^n$$

- Esta es la probabilidad de que Eve *no sea detectada*.
- **Probabilidad de Detección:**
  - Detectar a Eve = Ocurre al menos un error.

$$P(\text{Detección}) = 1 - P(\text{No detección})$$

$$P(\text{Detección}) = 1 - \left(\frac{3}{4}\right)^n$$

- **Conclusión:** Eligiendo n suficientemente grande, $P(\text{Detección}) \to 1$. (Ej: n=50, $P(\text{No detección}) \approx 5.6 \times 10^{-7}$)

Nota: Análisis simplificado. Pruebas formales más complejas.

Probabilidad de Detectar a Eve (Boceto - Parte 2)

Continuando con n bits comparados e independencia:

- Probabilidad de que Eve *no* introduzca error en *ninguno* de los n bits:

$$P(\text{Ningún error en } n \text{ bits}) = \left(\frac{3}{4}\right)^n$$

- Esta es la probabilidad de que Eve *no sea detectada*.
- **Probabilidad de Detección:**
  - Detectar a Eve = Ocurre al menos un error.

$$P(\text{Detección}) = 1 - P(\text{No detección})$$

$$P(\text{Detección}) = 1 - \left(\frac{3}{4}\right)^n$$

- **Conclusión:** Eligiendo n suficientemente grande, $P(\text{Detección}) \to 1$. (Ej: n=50, $P(\text{No detección}) \approx 5.6 \times 10^{-7}$)

Nota: Análisis simplificado. Pruebas formales más complejas.

Probabilidad de Detectar a Eve (Boceto - Parte 2)

Continuando con n bits comparados e independencia:

- Probabilidad de que Eve *no* introduzca error en *ninguno* de los n bits:

$$P(\text{Ningún error en } n \text{ bits}) = \left(\frac{3}{4}\right)^n$$

- Esta es la probabilidad de que Eve *no sea detectada*.
- **Probabilidad de Detección:**
  - Detectar a Eve = Ocurre al menos un error.

$$P(\text{Detección}) = 1 - P(\text{No detección})$$

$$P(\text{Detección}) = 1 - \left(\frac{3}{4}\right)^n$$

- **Conclusión:** Eligiendo n suficientemente grande, $P(\text{Detección}) \to 1$. (Ej: n=50, $P(\text{No detección}) \approx 5.6 \times 10^{-7}$)

Nota: Análisis simplificado. Pruebas formales más complejas.

Reconciliación de Información (Parte 1)

Incluso sin Eve, la clave cruda (post-sifting) puede tener errores:

- Ruido en el canal cuántico.
- Imperfecciones en detectores o fuentes.

El QBER estimado (δ) nos da una medida de estos errores.

Objetivo: Alice y Bob deben asegurarse de tener *exactamente* la misma clave.

Método: Usan un protocolo de *corrección de errores* sobre un canal *público clásico*.

- Intercambian información sobre sus claves (e.g., paridad de bloques de bits) para localizar y corregir discrepancias.
- Protocolos comunes: Cascade, Winnow.

Reconciliación de Información (Parte 1)

Incluso sin Eve, la clave cruda (post-sifting) puede tener errores:

- Ruido en el canal cuántico.
- Imperfecciones en detectores o fuentes.

El QBER estimado (δ) nos da una medida de estos errores.

Objetivo: Alice y Bob deben asegurarse de tener *exactamente* la misma clave.

Método: Usan un protocolo de *corrección de errores* sobre un canal *público clásico*.

- Intercambian información sobre sus claves (e.g., paridad de bloques de bits) para localizar y corregir discrepancias.
- Protocolos comunes: Cascade, Winnow.

Reconciliación de Información (Parte 1)

Incluso sin Eve, la clave cruda (post-sifting) puede tener errores:

- Ruido en el canal cuántico.
- Imperfecciones en detectores o fuentes.

El QBER estimado (δ) nos da una medida de estos errores.

Objetivo: Alice y Bob deben asegurarse de tener *exactamente* la misma clave.

Método: Usan un protocolo de *corrección de errores* sobre un canal *público clásico*.

- Intercambian información sobre sus claves (e.g., paridad de bloques de bits) para localizar y corregir discrepancias.
- Protocolos comunes: Cascade, Winnow.

Reconciliación de Información (Parte 2)

Aspectos importantes de la Reconciliación:

- La información intercambiada públicamente (ej. paridades) *no debe* revelar la clave completa. ¡Debe diseñarse cuidadosamente!
- Sin embargo, esta comunicación pública *sí filtra algo de información* sobre la clave a Eve.
    - Eve escucha la discusión sobre paridades, etc.
    - La cantidad de información filtrada es (aproximadamente) la cantidad de bits intercambiados.

Resultado de la Reconciliación:

- Alice y Bob ahora comparten una clave idéntica k'.
- Pero... Eve podría tener alguna información parcial sobre k'.

Reconciliación de Información (Parte 2)

Aspectos importantes de la Reconciliación:

- La información intercambiada públicamente (ej. paridades) *no debe* revelar la clave completa. ¡Debe diseñarse cuidadosamente!
- Sin embargo, esta comunicación pública *sí filtra algo de información* sobre la clave a Eve.
  - Eve escucha la discusión sobre paridades, etc.
  - La cantidad de información filtrada es (aproximadamente) la cantidad de bits intercambiados.

Resultado de la Reconciliación:

- Alice y Bob ahora comparten una clave idéntica k'.
- Pero... Eve podría tener alguna información parcial sobre k'.

Reconciliación de Información (Parte 2)

Aspectos importantes de la Reconciliación:

- La información intercambiada públicamente (ej. paridades) *no debe* revelar la clave completa. ¡Debe diseñarse cuidadosamente!
- Sin embargo, esta comunicación pública *sí filtra algo de información* sobre la clave a Eve.
  - Eve escucha la discusión sobre paridades, etc.
  - La cantidad de información filtrada es (aproximadamente) la cantidad de bits intercambiados.

Resultado de la Reconciliación:

- Alice y Bob ahora comparten una clave idéntica k'.
- Pero... Eve podría tener alguna información parcial sobre k'.

Amplificación de Privacidad (Parte 1)

Situación post-reconciliación:

- Alice y Bob tienen la misma clave k'.
- Eve tiene información parcial sobre k' debido a:
  - La comunicación pública durante la reconciliación.
  - Potencialmente, sus mediciones iniciales si no fue detectada (aunque la probabilidad sea baja).

Objetivo: Reducir la información de Eve sobre la clave final a un nivel negligible (arbitrariamente pequeño).

Método: Aplican una *función hash universal*.

- Eligen (al azar de una familia conocida) y comunican públicamente qué función hash usarán.
- Aplican esa función a su clave reconciliada k':

$$k_{\text{final}} = \text{Hash}(k')$$

Amplificación de Privacidad (Parte 1)

Situación post-reconciliación:

- Alice y Bob tienen la misma clave k'.
- Eve tiene información parcial sobre k' debido a:
  - La comunicación pública durante la reconciliación.
  - Potencialmente, sus mediciones iniciales si no fue detectada (aunque la probabilidad sea baja).

Objetivo: Reducir la información de Eve sobre la clave final a un nivel negligible (arbitrariamente pequeño).

Método: Aplican una *función hash universal*.

- Eligen (al azar de una familia conocida) y comunican públicamente qué función hash usarán.
- Aplican esa función a su clave reconciliada k':

$$k_{\text{final}} = \text{Hash}(k')$$

Amplificación de Privacidad (Parte 1)

Situación post-reconciliación:

- Alice y Bob tienen la misma clave k'.
- Eve tiene información parcial sobre k' debido a:
  - La comunicación pública durante la reconciliación.
  - Potencialmente, sus mediciones iniciales si no fue detectada (aunque la probabilidad sea baja).

Objetivo: Reducir la información de Eve sobre la clave final a un nivel negligible (arbitrariamente pequeño).

Método: Aplican una *función hash universal*.

- Eligen (al azar de una familia conocida) y comunican públicamente qué función hash usarán.
- Aplican esa función a su clave reconciliada k':

$$k_{\text{final}} = \text{Hash}(k')$$

Amplificación de Privacidad (Parte 2)

Idea detrás del Hashing Universal:

- Comprimen la clave k' (de longitud N') a una clave final k_{final} más corta (de longitud $N_{\text{final}} < N'$).
- La cantidad de compresión ($N' - N_{\text{final}}$) se elige cuidadosamente. Depende de:
  - Cuánta información se estima que Eve pudo obtener (relacionado con QBER).
  - Cuánta información se filtró durante la reconciliación.
- Aunque Eve conozca la función hash, la salida k_{final} le revela muy poca información sobre la entrada k', especialmente si la compresión es suficiente.

Resultado de la Amplificación de Privacidad:

- Obtienen una clave final k_{final} más corta.
- La información de Eve sobre k_{final} es *arbitrariamente pequeña* (decrece exponencialmente con la compresión $N' - N_{\text{final}}$).

¡Ahora sí, Alice y Bob comparten una clave secreta y segura!

Amplificación de Privacidad (Parte 2)

Idea detrás del Hashing Universal:

- Comprimen la clave k' (de longitud N') a una clave final k_{final} más corta (de longitud $N_{\mathrm{final}} < N'$).
- La cantidad de compresión ($N' - N_{\mathrm{final}}$) se elige cuidadosamente. Depende de:
    - Cuánta información se estima que Eve pudo obtener (relacionado con QBER).
    - Cuánta información se filtró durante la reconciliación.
- Aunque Eve conozca la función hash, la salida k_{final} le revela muy poca información sobre la entrada k', especialmente si la compresión es suficiente.

Resultado de la Amplificación de Privacidad:

- Obtienen una clave final k_{final} más corta.
- La información de Eve sobre k_{final} es *arbitrariamente pequeña* (decrece exponencialmente con la compresión $N' - N_{\mathrm{final}}$).

¡Ahora sí, Alice y Bob comparten una clave secreta y segura!

Amplificación de Privacidad (Parte 2)

Idea detrás del Hashing Universal:

- Comprimen la clave k' (de longitud N') a una clave final k_{final} más corta (de longitud $N_{\text{final}} < N'$).
- La cantidad de compresión ($N' - N_{\text{final}}$) se elige cuidadosamente. Depende de:
  - Cuánta información se estima que Eve pudo obtener (relacionado con QBER).
  - Cuánta información se filtró durante la reconciliación.
- Aunque Eve conozca la función hash, la salida k_{final} le revela muy poca información sobre la entrada k', especialmente si la compresión es suficiente.

Resultado de la Amplificación de Privacidad:

- Obtienen una clave final k_{final} más corta.
- La información de Eve sobre k_{final} es *arbitrariamente pequeña* (decrece exponencialmente con la compresión $N' - N_{\text{final}}$).

¡Ahora sí, Alice y Bob comparten una clave secreta y segura!

Amplificación de Privacidad (Parte 2)

Idea detrás del Hashing Universal:

- Comprimen la clave k' (de longitud N') a una clave final k_{final} más corta (de longitud $N_{\text{final}} < N'$).
- La cantidad de compresión ($N' - N_{\text{final}}$) se elige cuidadosamente. Depende de:
  - Cuánta información se estima que Eve pudo obtener (relacionado con QBER).
  - Cuánta información se filtró durante la reconciliación.
- Aunque Eve conozca la función hash, la salida k_{final} le revela muy poca información sobre la entrada k', especialmente si la compresión es suficiente.

Resultado de la Amplificación de Privacidad:

- Obtienen una clave final k_{final} más corta.
- La información de Eve sobre k_{final} es *arbitrariamente pequeña* (decrece exponencialmente con la compresión $N' - N_{\text{final}}$).

¡Ahora sí, Alice y Bob comparten una clave secreta y segura!

Amplificación de Privacidad (Parte 2)

Idea detrás del Hashing Universal:

- Comprimen la clave k' (de longitud N') a una clave final k_{final} más corta (de longitud $N_{\text{final}} < N'$).
- La cantidad de compresión ($N' - N_{\text{final}}$) se elige cuidadosamente. Depende de:
  - Cuánta información se estima que Eve pudo obtener (relacionado con QBER).
  - Cuánta información se filtró durante la reconciliación.
- Aunque Eve conozca la función hash, la salida k_{final} le revela muy poca información sobre la entrada k', especialmente si la compresión es suficiente.

Resultado de la Amplificación de Privacidad:

- Obtienen una clave final k_{final} más corta.
- La información de Eve sobre k_{final} es *arbitrariamente pequeña* (decrece exponencialmente con la compresión $N' - N_{\text{final}}$).

¡Ahora sí, Alice y Bob comparten una clave secreta y segura!

Resumen del Proceso QKD Completo (Parte 1)

1. **Fase Cuántica:**
   - Alice envía qubits (preparados en bases Z o X al azar).
   - Bob mide (en bases Z o X al azar).
   - (Usando protocolo BB84, E91, etc.)
2. **Sifting (Cernido):**
   - A y B publican bases usadas (por canal clásico).
   - Descartan resultados donde usaron bases distintas.
   - Resultado: *Clave Cruda* (Raw Key).
3. **Parameter Estimation (Estimación de QBER):**
   - A y B comparan públicamente una fracción aleatoria de la Clave Cruda.
   - Calculan QBER (δ).
   - Si δ es muy alto $\implies$ Abortar (posible Eve).

Resumen del Proceso QKD Completo (Parte 1)

1. **Fase Cuántica:**
   - Alice envía qubits (preparados en bases Z o X al azar).
   - Bob mide (en bases Z o X al azar).
   - (Usando protocolo BB84, E91, etc.)
2. **Sifting (Cernido):**
   - A y B publican bases usadas (por canal clásico).
   - Descartan resultados donde usaron bases distintas.
   - Resultado: *Clave Cruda* (Raw Key).
3. **Parameter Estimation (Estimación de QBER):**
   - A y B comparan públicamente una fracción aleatoria de la Clave Cruda.
   - Calculan QBER (δ).
   - Si δ es muy alto $\implies$ Abortar (posible Eve).

Resumen del Proceso QKD Completo (Parte 1)

1. **Fase Cuántica:**
   - Alice envía qubits (preparados en bases Z o X al azar).
   - Bob mide (en bases Z o X al azar).
   - (Usando protocolo BB84, E91, etc.)
2. **Sifting (Cernido):**
   - A y B publican bases usadas (por canal clásico).
   - Descartan resultados donde usaron bases distintas.
   - Resultado: *Clave Cruda* (Raw Key).
3. **Parameter Estimation (Estimación de QBER):**
   - A y B comparan públicamente una fracción aleatoria de la Clave Cruda.
   - Calculan QBER (δ).
   - Si δ es muy alto $\implies$ Abortar (posible Eve).

Resumen del Proceso QKD Completo (Parte 2)

4. **Reconciliation (Corrección de Errores):**
   - A y B usan comunicación pública para corregir errores en la Clave Cruda restante.
   - Protocolos como Cascade.
   - Resultado: *Clave Reconciliada* (idéntica para A y B, pero parcialmente conocida por Eve).
5. **Privacy Amplification (Amplificación de Privacidad):**
   - A y B aplican hashing universal a la Clave Reconciliada.
   - Comprimen la clave para eliminar la información de Eve.
   - Resultado: *Clave Final Segura*.

Resultado Final: Una clave secreta compartida, cuya seguridad se basa en principios cuánticos.

Resumen del Proceso QKD Completo (Parte 2)

4. **Reconciliation (Corrección de Errores):**
   - A y B usan comunicación pública para corregir errores en la Clave Cruda restante.
   - Protocolos como Cascade.
   - Resultado: *Clave Reconciliada* (idéntica para A y B, pero parcialmente conocida por Eve).
5. **Privacy Amplification (Amplificación de Privacidad):**
   - A y B aplican hashing universal a la Clave Reconciliada.
   - Comprimen la clave para eliminar la información de Eve.
   - Resultado: *Clave Final Segura*.

Resultado Final: Una clave secreta compartida, cuya seguridad se basa en principios cuánticos.

Resumen del Proceso QKD Completo (Parte 2)

4. **Reconciliation (Corrección de Errores):**
   - A y B usan comunicación pública para corregir errores en la Clave Cruda restante.
   - Protocolos como Cascade.
   - Resultado: *Clave Reconciliada* (idéntica para A y B, pero parcialmente conocida por Eve).
5. **Privacy Amplification (Amplificación de Privacidad):**
   - A y B aplican hashing universal a la Clave Reconciliada.
   - Comprimen la clave para eliminar la información de Eve.
   - Resultado: *Clave Final Segura*.

Resultado Final: Una clave secreta compartida, cuya seguridad se basa en principios cuánticos.

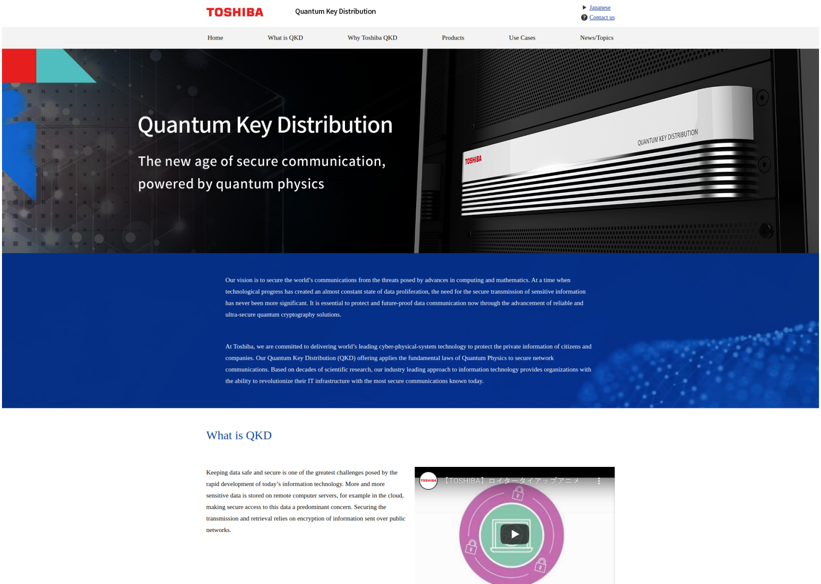
BB84

El protocolo BB84 es similar pero usa cuatro estados $\left\{ |0\rangle, |1\rangle, |+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), |-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \right\}$. Lo que Alice y Bob publican es la base de preparación y la de medición.

E91

El protocolo E91 usa entrelazamiento, pero termina siendo reducible al BB84. Alice y Bob comparten el estado $|\Psi-\rangle$ de la base de Bell. Miden ambos en una base al azar entre B_1 y B_2. Los resultados, cuando miden en la misma base, están anticorrelacionados.
Pizarra.

¿Esto existe?


TOSHIBA
Quantum Key Distribution
Japanese
Contact us
Home
What is QKD
Why Toshiba QKD
Products
Use Cases
News/Topics
Quantum Key Distribution
The new age of secure communication, powered by quantum physics
TOSHIBA
QUANTUM KEY DISTRIBUTION
Our vision is to secure the world's communications from the threats posed by advances in computing and mathematics. At a time when technological progress has created an almost constant state of data proliferation, the need for the secure transmission of sensitive information has never been more significant. It is essential to protect and future-proof data communication now through the advancement of reliable and ultra-secure quantum cryptography solutions.
At Toshiba, we are committed to delivering world's leading cyber-physical-system technology to protect the private information of citizens and companies. Our Quantum Key Distribution (QKD) offering applies the fundamental laws of Quantum Physics to secure network communications. Based on decades of scientific research, our industry leading approach to information technology provides organizations with the ability to revolutionize their IT infrastructure with the most secure communications known today.
What is QKD
Keeping data safe and secure is one of the greatest challenges posed by the rapid development of today's information technology. More and more sensitive data is stored on remote computer servers, for example in the cloud, making secure access to this data a predominant concern. Securing the transmission and retrieval relies on encryption of information sent over public networks.

Vimos hoy

- Algoritmos de Deutsch y Deutsch-Jozsa.
- QKD.

La que viene

Factorización.