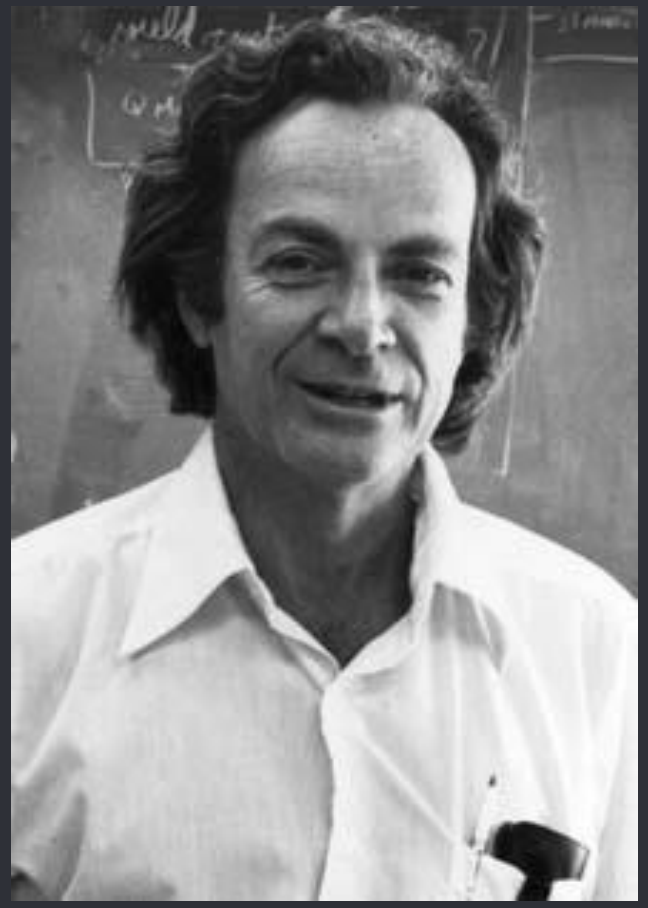
Computación Cuántica

¿De donde sale la idea de una computadora cuántica?

Richard P. Feynman mediante su articulo llamado “Simulating Physics with Computers” propuso que simular física cuántica era necesario construir computadoras cuánticas para describir de una manera mas precisa lo que ocurre a esas escalas.

Propone algunos experimentos que para ser resueltos de manera exacta no bastaría con simplemente ir escalando una computadora o aplicando un paralelismo masivo ya que estos sistemas contienen muchos estados posibles.



Fotografía de Richard Feynman

Otros avances teóricos importantes:

- 1985 - David Deutsch describió el primer computador cuántico universal
- 1993 Dan Simon demostraba la ventaja que tendría un computadora cuántica frente a una tradicional al comparar el modelo de probabilidad clásica con el modelo cuántico.
- Entre 1994 y 1995 Peter Shor definió el algoritmo que lleva su nombre y que permite calcular los factores primos de números a una velocidad mucho mayor que en cualquier computadora tradicional.
- En 1996 Lov Grover propone el algoritmo de búsqueda de datos que lleva su nombre. Al igual que el resto de algoritmos cuánticos, se trata de un algoritmo probabilístico con un alto índice de acierto.

Vamos a tratar de entender que hay detrás de todo esto.

Mecánica Cuántica

Es el estudio de fenómenos a escala microscópica mediante las hipótesis de la cuantización de la energía y la dualidad onda-partícula.

Una de sus interpretaciones (Interpretación de Copenhague y la mas aceptada) nos dice que un estado cuántico esta descrito por una función de onda que describe la probabilidad de que dicha partícula este en un estado u otro.

Propiedades que nos interesan:

- Superposición.
- Entrelazamiento.
- Decoherencia.

Estado cuántico

Podemos definir un estado cuántico como el estado físico en un momento dado de un sistema.

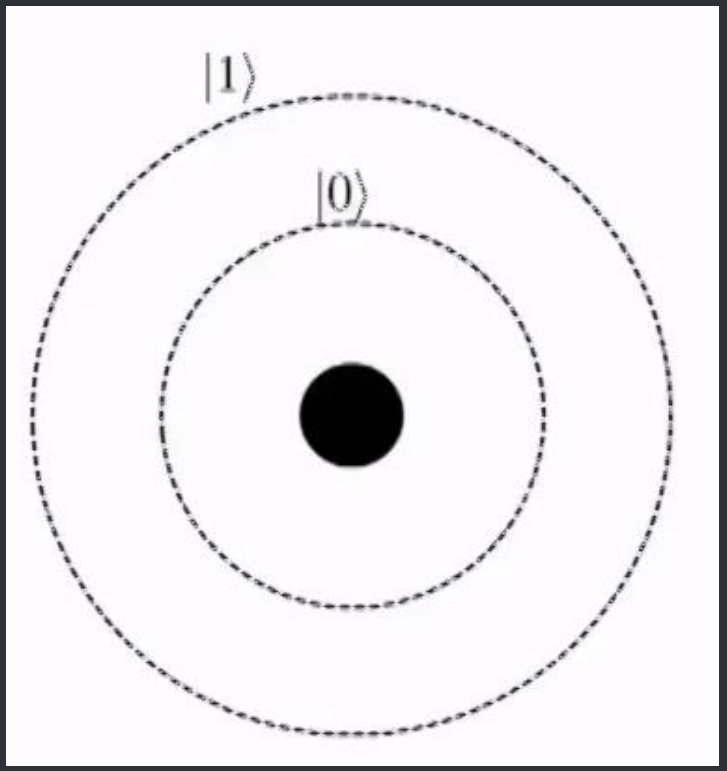
Podemos describir un estado cuántico con los niveles de energía de un electrón en un átomo de baja densidad.

Al estado 1 se le conoce como

“exited state”

Y al estado 0 como

“ground state”

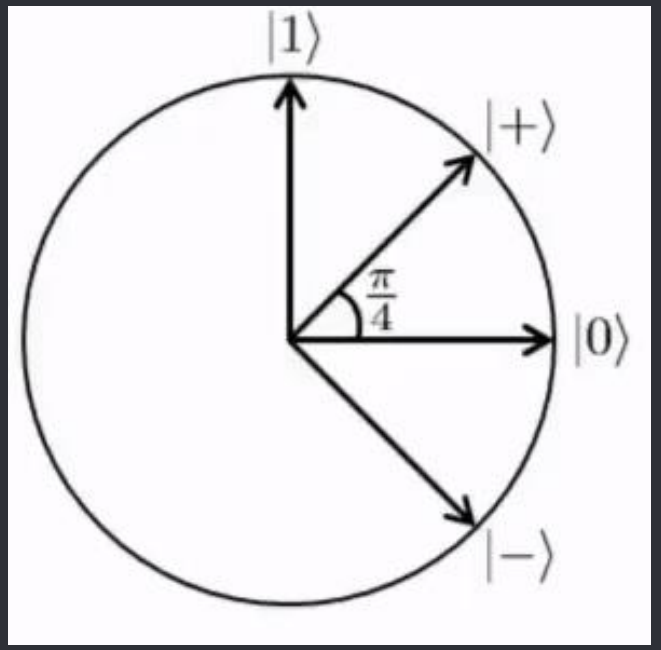


También se puede trabajar con la propiedad del spin ya que con campos magnéticos se puede inducir que el spin del electrón quede en una dirección.

Superposición

Nos dice que un estado cuántico no esta definido por completo y puede estar en un estado de superposición ya que al ser una teoría lineal podemos tener estados como

$$|\psi\rangle = \alpha_0|0\rangle + \alpha_1|1\rangle = \beta_0|+\rangle + \beta_1|-\rangle$$



Entrelazamiento

Esto quiere decir que cuando dos o mas partículas partículas están entrelazadas cuando queremos medir alguna propiedad, se va a medir el estado del sistema y no el de una partícula individual.

Esto trae consigo que ambas partículas al ser medidas colapsen y se tenga un estado en el que una partícula depende de otra.

Esto tiene implicaciones por ejemplo, para la transmisión de información.

Decoherencia

Cuando un observador mide alguna propiedad de una partícula, provoca un colapso de la función de onda de dicha partícula.

Ésta va a tener una probabilidad de colapsar en cierto estado según cómo este dado dicho estado.

$$p_0 = \sum_{\substack{0 \le x < 2^n \\ x_k = 0}} |a_x|^2$$

¿De que nos sirve conocer esto?

Sabemos que en una computadora clásica podemos representar la información en bits (0,1) y estos van a estar bien definidos.

Para términos cuánticos tenemos los llamados Qbits que al igual que los bits tradicionales éstos pueden estar en 0 o 1 pero añadiendo el estado de superposición podemos estar en alguno de esos dos estados con diferentes coeficientes. Puede estar mas de 1 que de 0 o viceversa.

Con estos estados podemos representar la información.

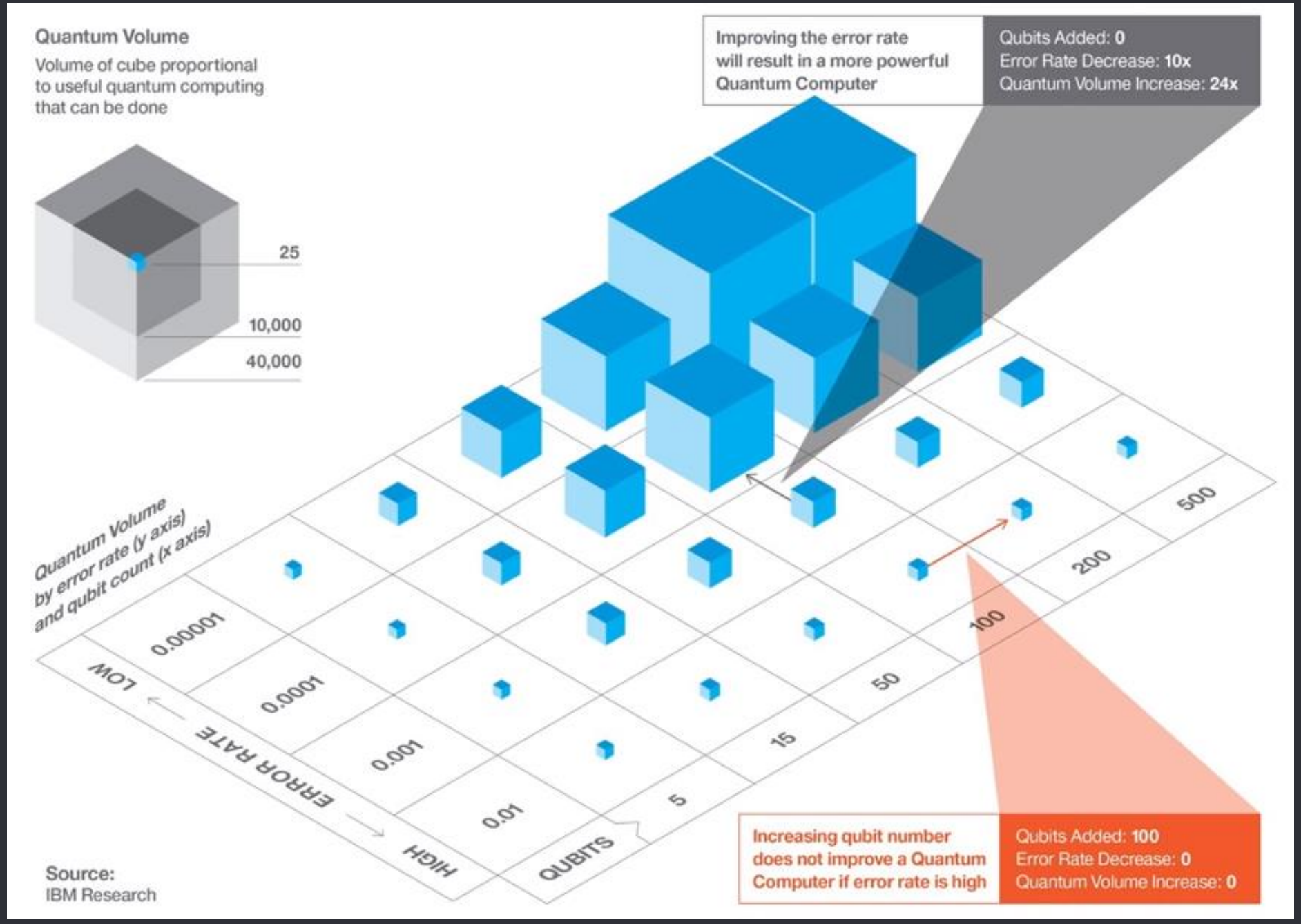
Si queremos hacer una función con n bits, lo que hacemos es ir operándolos uno a uno para poder obtener las diferentes salidas según la función que tengamos.

Un Qbit puede representar la misma información que un bit pero la diferencia radica en los estados de superposición que se pueden evaluar simultáneamente en la entrada y con compuertas cuánticas ir modificando la probabilidad de estos estados para maximizar el resultado esperado.

$$\alpha_{00}|00\rangle + \alpha_{01}|01\rangle + \alpha_{10}|10\rangle + \alpha_{11}|11\rangle$$

$$\alpha_{000}|000\rangle + \alpha_{001}|001\rangle + \alpha_{010}|010\rangle + \alpha_{011}|011\rangle + \cdots + \alpha_{111}|111\rangle$$

No debemos olvidar que es una teoría probabilística.


Quantum Volume
Volume of cube proportional to useful quantum computing that can be done
25
10,000
40,000
Improving the error rate will result in a more powerful Quantum Computer
Qubits Added: 0
Error Rate Decrease: 10x
Quantum Volume Increase: 24x
Quantum Volume by error rate (y axis) and qubit count (x axis)
ERROR RATE
LOW
HIGH
0.00001
0.0001
0.001
0.01
QUBITS
5
15
50
100
200
500
Increasing qubit number does not improve a Quantum Computer if error rate is high
Qubits Added: 100
Error Rate Decrease: 0
Quantum Volume Increase: 0
Source:
IBM Research

¿Cómo funcionan los algoritmos?

Como en la electrónica digital, existen diferentes tipos de compuertas que nos permiten hacer una transformación unitaria del estado cuántico (matemáticamente una matriz) la cual nos permite describir la evolución de este estado cuántico. Algunos ejemplos son

$$U_{NOT} = \sigma_1 = X \equiv \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \qquad |0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Compuerta Not.

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = \frac{1}{\sqrt{2}}(X + Z)$$

Compuerta Hadamard.

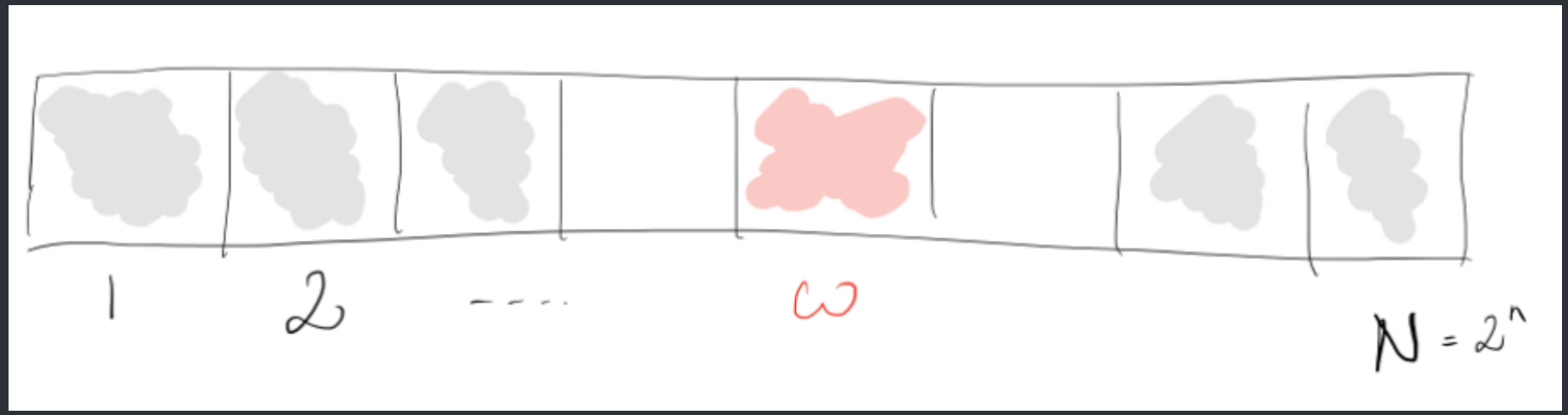
Algoritmo de Búsqueda de Grover

Este algoritmo puede acelerar un problema de búsqueda no estructurada de forma cuadrática, pero sus usos se extienden más allá de eso; Puede servir como un truco general para obtener mejoras de tiempo de ejecución cuadráticas para una variedad de otros algoritmos.

Esto se llama el truco de amplificación de amplitud.

Búsqueda no estructurada

Supongamos que tenemos una lista de tamaño N y tenemos un elemento único en la lista que queremos encontrar. La lista no tiene ningún orden especifico por lo que tenemos algo como esto.



La solución mas inmediata es comparando cada elemento con la propiedad que estamos buscando, en promedio, su complejidad va a ser de N/2 y en el peor de los casos tenemos que se debe comparar toda la lista. Pero el algoritmo de grover lo hace en $\sqrt{N}$ pero ¿como lo hace?

Lo primero que hacemos es codificar la información, para los ítems que no son los que buscamos definimos una función f(x) =0 mientras que para el ítem buscado será f(w)=1

Para usar una computadora cuántica para este problema, debemos proporcionar los elementos en superposición a esta función, por lo que codificamos la función en una matriz unitaria llamada oráculo.

Primero elegimos una codificación binaria de los ítems x, w∈ {0,1}

Ahora definimos la matriz que nos da alguno de los estados que tenemos definidos para x.

$$U_f|x\rangle = (-1)^{f(x)}|x\rangle.$$

Como vemos, cuando evaluamos x estamos en el mismo estado mientras que cuando evaluamos w (que es el ítem que queremos encontrar) la función mapea a $U_f|w\rangle = -|w\rangle$

Entonces, ¿cómo funciona el algoritmo?

Antes de mirar la lista de elementos, no tenemos idea de dónde está el elemento marcado. Por lo tanto, cualquier conjetura de su ubicación es tan buena como cualquier otra, que se puede expresar en términos de un estado cuántico llamado superposición uniforme:

$$|s\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle.$$

Si en este punto midiéramos en la base estándar {| x⟩}, esta superposición colapsaría en cualquiera de los estados base con la misma probabilidad de

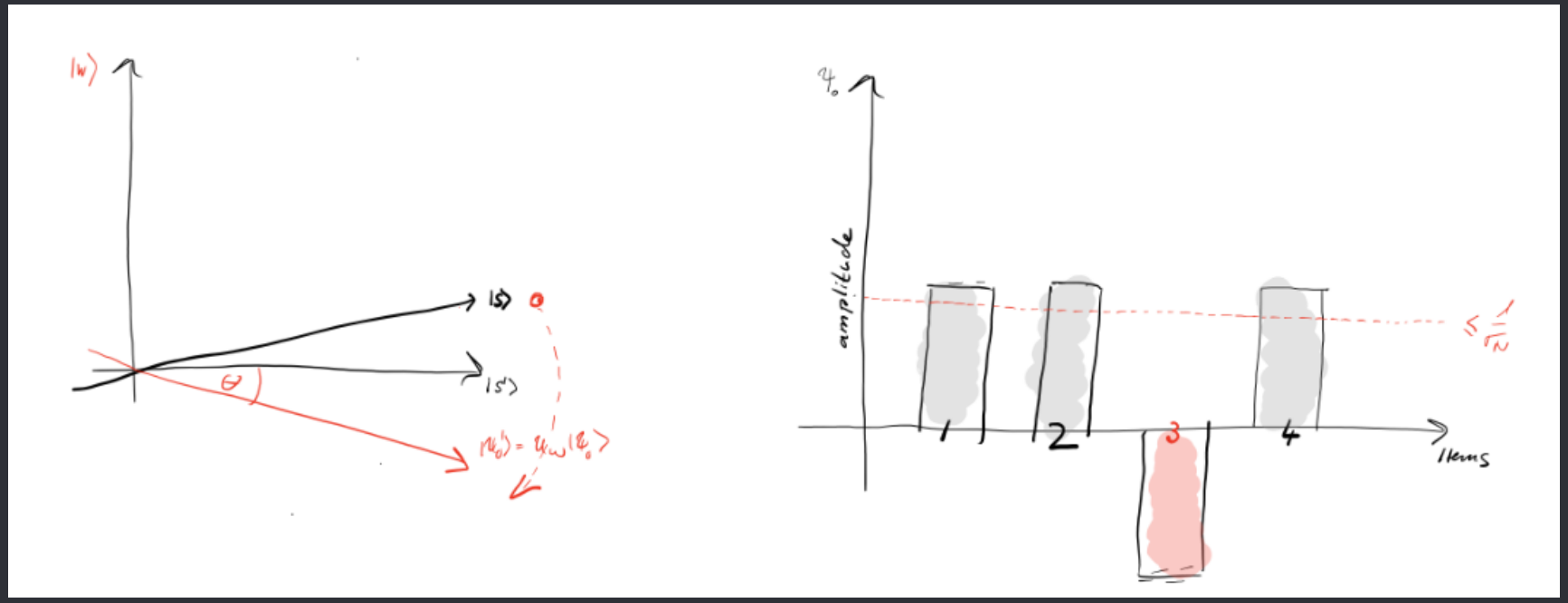
$$\frac{1}{N} = \frac{1}{2^n}.$$

Lo que queremos es amplificar la amplitud de el ítem que queremos encontrar por lo que

Paso 0. Partimos de una superposición uniforme. Ésta se puede realizar mediante la compuerta de Hadamard.

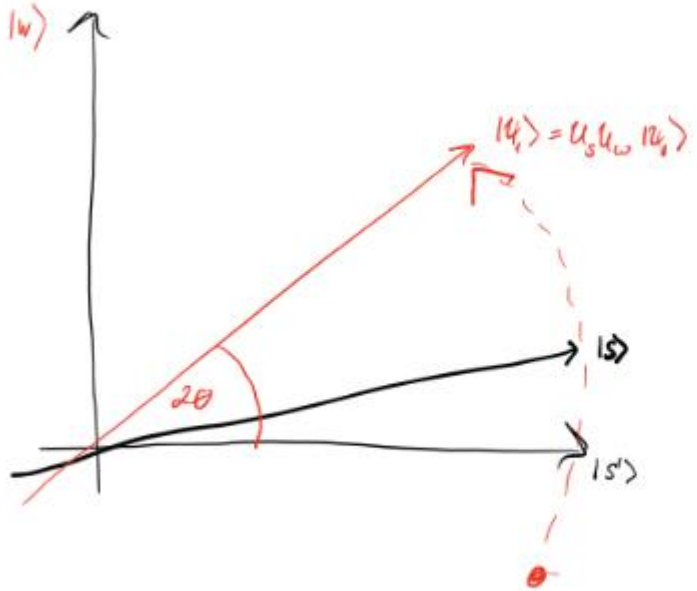
$$|s\rangle = H^{\otimes n}|0\rangle^n$$

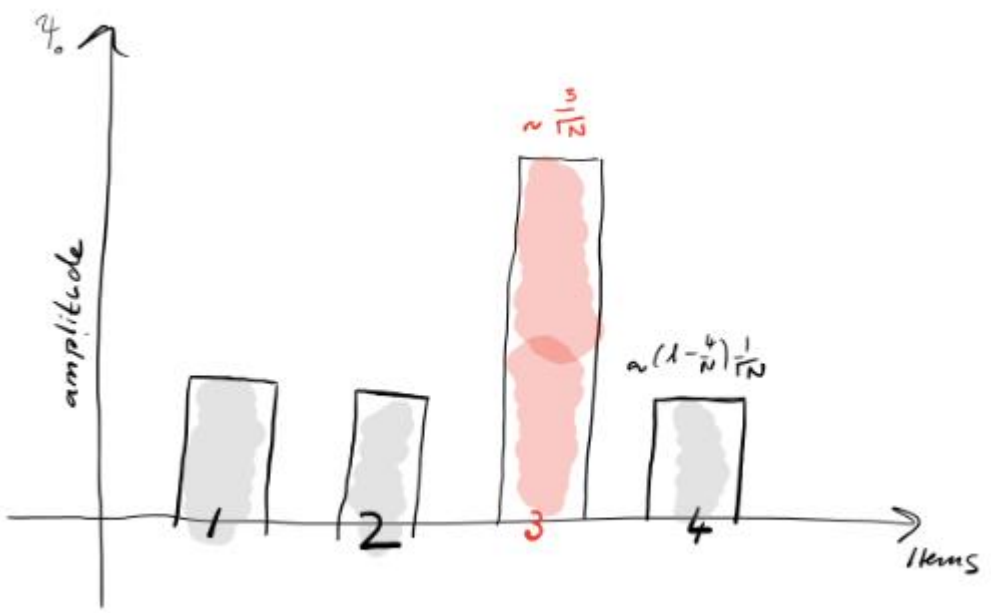
Paso 1. Aplicamos una reflexión de fase



Paso 2. Aplicamos otra rotación de fase, pero ahora en sentido inverso por lo que tenemos

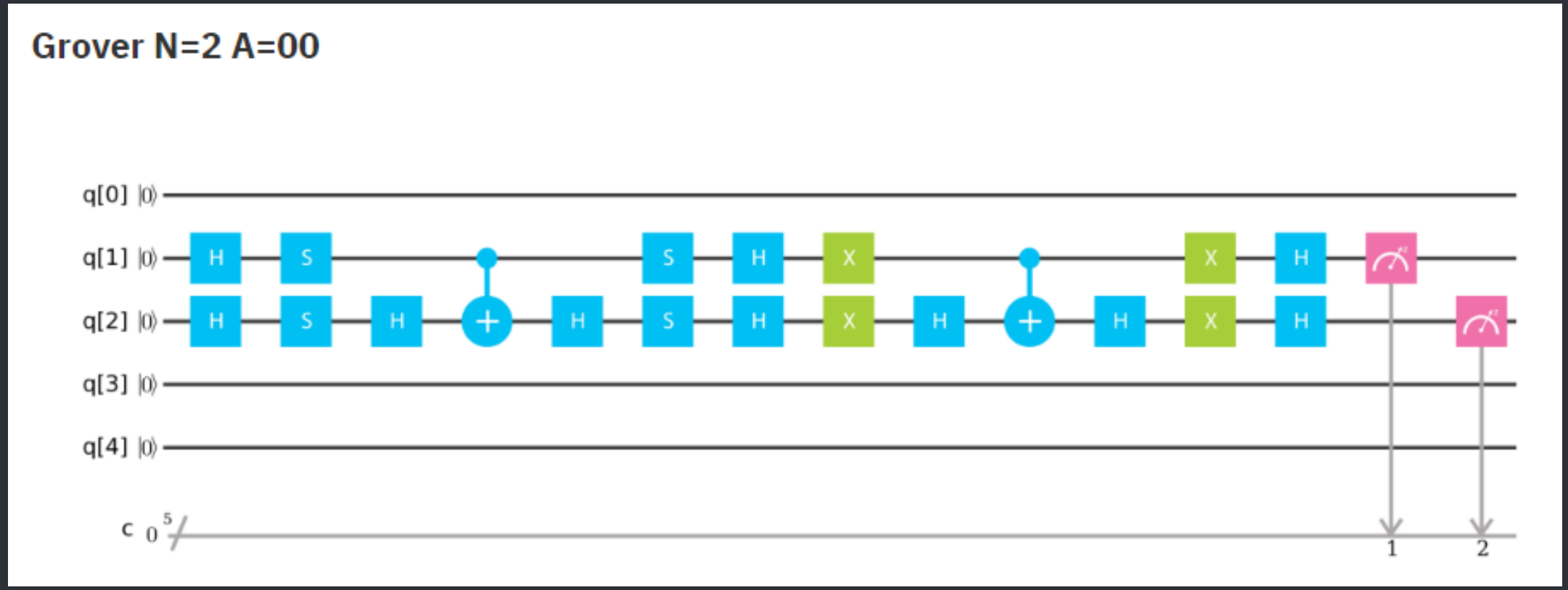
$$|\psi_{t+1}\rangle = U_s U_f |\psi_t\rangle$$





Si este proceso lo repetimos varias veces, tenemos que la probabilidad aumenta para el ítem que queremos encontrar mientras que para los otros disminuye.

Este proceso lo tenemos que hacer $\sqrt{N}$ aproximadamente, finalmente lo podemos representar en IBM Quantum Experience



Referencias

https://thediplomatinspain.com/2018/05/conferencia-en-el-instituto-internacional-sobre-richard-feynman/

A Beginner’s Guide to Quantum Computing https://www.youtube.com/watch?v=S52rxZG-zi0

https://www.nucleares.unam.mx/~vieyra/node23.html

http://www.criptored.upm.es/crypt4you/temas/cuantica/leccion1/leccion01.html

Richard P. Feynman. (1982). Simulating Physics with Computers . 6 de abril de 2019, de International Journal of Theoretical Physic Sitio web: https://people.eecs.berkeley.edu/~christos/classics/Feynman.pdf

https://quantumexperience.ng.bluemix.net/proxy/tutorial/full-user-guide/004-Quantum_Algorithms/070-Grover's_Algorithm.html

https://www.youtube.com/watch?v=Z1uoz_8dLH0&list=PL74Rel4IAsETUwZS_Se_P-fSEyEVQwni7