Computación Cuántica

Chenjie Huang

Índice

1. Introducción **2**
1.1. Objetivos . . . 2
1.2. Metodología . . . 3

2. Preliminares **4**
2.1. Espacios Vectoriales . . . 4
2.2. Bases y dimensión . . . 4
2.3. Aplicaciones lineales y forma matricial . . . 5
2.4. Producto Escalar y Espacios de Hilbert . . . 6
2.5. Matrices Adjuntas o Hermitianas . . . 7
2.6. Producto Tensorial . . . 8

3. Estructuras y Puertas Cuánticas **10**
3.1. Qubit . . . 10
3.2. Sistema de Varios Qubits . . . 11
3.3. Circuitos y Puertas Cuánticas . . . 12

4. Algoritmos Cuánticos **18**
4.1. Algoritmo de Teletransportación . . . 18
4.2. Algoritmo de Deutsch . . . 19
4.3. Algoritmo de Deutsch-Jozsa . . . 22
4.4. Algoritmo de Búsqueda de Grover . . . 26
4.5. Algoritmo de Periodicidad de Simon . . . 29
4.6. Algoritmo de Factorización de Shor . . . 31

5. Conclusión **38**
5.1. Abstract . . . 38

1. Introducción

La computación cuántica es un área de estudio reciente y en desarrollo que implica otras grandes áreas de estudio como la computación, la física y la matemática. Dicho estudio se centra en los curiosos fenómenos y aspectos de la física cuántica, todo con el objetivo de expandir los límites de la computación. Esto nos ha permitido dar algunas soluciones innovadoras a problemas complejos de la computación sin resolver y desde un punto de vista totalmente nuevo.

La mecánica cuántica, base de nuestro campo que es la computación cuántica, es de las ciencias con más éxito pero a la vez desconocida. Empezando su desarrollo teórico en el siglo pasado, seguido de su su aplicación por los científicos en la observación y entendimiento de los fundamentos de las partículas y sus comportamientos, resultando incluso en el desarrollo de un modelo físico que es la física de partículas. A pesar de su aplicaciones para explicar un amplio rango de fenómenos de la naturaleza, un leve cambio en esta idea de aplicar los sistemas cuánticos en los fenómenos naturales para darles explicación da lugar a la idea de no limitarse a observar, sino a diseñar estos sistemas cuánticos. Como resultado de esto, surge un interés de nuevo por la mecánica cuántica, sobretodo en planteamiento de cuestiones que llegan a combinar otros campos de estudio como la computación y la teoría de la información.

Como veremos en detalle en poco, la idea tras la computación cuántica es aprovechar el fenómeno de la superposición cuántica, descrito en la teoría de la mecánica cuántica. Este fenómeno describe partículas que pueden estar en diferentes estados de manera simultanea, que determinaremos un estado fijo en la partícula en cuanto la observemos. Además permite interacciones con el elemento en varios estados de manera también simultanea. A pesar de que se desconoce la naturaleza y el porqué de este fenómeno, no podemos negar que esto ocurre e incluso dando lugar efectos sobre fenómenos reales. Podemos destacar en cuestión el experimento de la doble rendija, en el que pone en manifiesto el fenómeno de la superposición de varios estados, cuando el estado de un mismo elemento interacciona con otro estado de manera que a veces llega a .ªnularlo". A esto se le conoce como interferencia, y podemos poner en similitud un ejemplo de dos guijarros tirados a un charco de agua y cómo se forma ondas en la superficie de manera que a veces se llegan a cancelar unas a otras, mientras que otras veces se llegan a reforzar una a la otra.

En vista de la computación clásica, tenemos su elemento base que es el bit, un elemento con dos estados posibles y nos permite realizar operaciones sencillas pero y si las combinamos estas da lugar a numerosas tareas complejas que puede realizar un ordenador hoy en día. La computación cuántica intentará mejorar esto introduciendo el qubit ó *quantum bit*, que en vez de considerar dos estados posibles, consideraremos cualquier combinación de ellas en superposición dando lugar a infinitas combinaciones. Esto nos abrirá numerosas puertas en la computación, permitiendo por ejemplo la evaluación simultanea de una función sobre diferentes elementos, o más bien estados. Lo que nos da lugar a tareas que podremos realizar que se consideraban imposibles ya sea por su complejidad como por el tiempo requerido. Y no problemas sin importancia, pues uno de los temas que veremos es sobre la factorización de números, que es clave en la encriptografía, presente en más lugares de nuestra vida diaria de la que creemos.

1.1. Objetivos

Este trabajo de fin de grado tiene por propósito introducir al estudiante en este campo de estudio con una popularidad creciente en los últimos años. A pesar de tocar varios detalles del área que involucra conocimientos del campo de la física, de la computación y de la matemática, se ha centrado en estas dos últimas con especial interés en la matemática recogiendo nociones sobre espacios vectoriales y productos tensoriales, que será la base matemáticas para luego elaborar los algoritmos.

Como hemos mencionado antes, no entraremos en detalle en las cuestiones físicas, no es el interés de este trabajo, sino nos centraremos con las matemáticas en la que se basa el modelo físico, dando las representaciones físicas de los elementos matemáticos que estudiaremos con sólo el fin de proporcionar otro punto de visto. Pero en el fondo, lo trataremos como un objeto matemático a la hora de trabajar.

1.2. Metodología

Por tanto, con este objetivo en mente, se procederá de la siguiente manera. Se realizará la lectura de materia propuesto por el tutor en forma de dos libros. Uno de naturaleza más sencilla para dar una idea inicial al alumno, mientra que el otro de carácter más complejo servirá para entrar en detalles de cuestiones más complicadas además de a modo de consulta.

Se procederá entonces a la exposición y desarrollo de los conceptos que se han estudiado de los dos libros en las siguientes del documento. Esto conlleva claramente el entendimiento de estos conceptos y para ello se han realizado de manera constante numerosas tutorías ya sea para una duda puntual como para reafirmar y reforzar la idea presente en los libro. También ha servido estas tutorías como guía para organizar el trabajo y cómo llevarlo a cabo.

Además de la presentación en los contenidos de este documento, se ha procedido a la implementación de los algoritmos en Qiskit, un software que permite de manera relativamente sencilla programar los algoritmos cuánticos. Se han ejecutado simulaciones de los algoritmos además de algunas pruebas en un ordenador cuántico real. No mostraremos los códigos del algoritmo en este documento, sino los presentaremos en un repositorio abierto de GitHub.

Sin más preámbulos comenzaremos con el desarrollo del trabajo teórico.

2. Preliminares

La Teoría Cuántica se apoya principalmente sobre álgebra lineal, concretamente sobre el espacio vectorial complejo de dimensión finita $\mathbb{C}^n$. En esta sección que servirá como preliminares como indica el título, nos centraremos en la teoría de álgebra lineal sobre el espacio vectorial complejo.
El objetivo es conseguir que este apartado sirva a modo de fundamento y bases para secciones posteriores y también de consulta posteriormente.

2.1. Espacios Vectoriales

Definición 2.1. *Un espacio vectorial sobre un cuerpo $\mathbb{K}$ es un conjunto no vacío $\mathbb{V}$, cuyo elementos llamaremos vectores, y llevan asociado dos operaciones,*

- *La Suma, $+ : \mathbb{V} \times \mathbb{V} \longrightarrow \mathbb{V}$*
- *El Producto por un escalar, $\cdot : \mathbb{K} \times \mathbb{V} \longrightarrow \mathbb{V}$*

*tal que $(\mathbb{V}, +)$ cumple las propiedades de formar un **grupo abeliano** y el producto por un escalar $\cdot$ cumpla las propiedades de:*

- *Existencia de elemento neutro:*

$$\exists e \in \mathbb{K} \text{ tal que } \forall v \in \mathbb{V}, e \cdot v = v \tag{2.1}$$

- *Propiedad asociativa:*

$$\forall a, b \in \mathbb{K}, \forall v \in \mathbb{V}, a \cdot (b \cdot v) = (a \cdot b) \cdot v \tag{2.2}$$

- *Propiedad distributiva respecto a la suma de vectores:*

$$\forall a \in \mathbb{K}, \forall u, v \in \mathbb{V}, a \cdot (u + v) = a \cdot u + a \cdot v \tag{2.3}$$

- *Propiedad distributiva respecto a la suma de escalares:*

$$\forall a, b \in \mathbb{K}, \forall v \in \mathbb{V}, (a + b) \cdot v = a \cdot v + b \cdot v \tag{2.4}$$

En el caso de que el cuerpo de escalares sea el de los complejos $\mathbb{C}$, se le denominará **espacio vectorial complejo**, siendo estas de gran interés para nuestro campo de estudio que es la mecánica cuántica. A partir de ahora usaremos $\mathbb{C}$ como cuerpo de escalares del espacio vectorial junto a la notación estándar de mecánica cuántica para referirnos a los elementos básicos de la álgebra lineal.

Denotaremos al vector en un espacio vectorial $\mathbb{V}$ como $|v\rangle$, donde usaremos $|\cdot\rangle$ para indicar que es un vector del espacio, denominado ***ket***.
En cuanto al elemento neutro del espacio vectorial, el vector cero, lo denotaremos excepcionalmente como $\mathbf{0}$. Veremos posteriormente que usaremos $|0\rangle$ para referirnos a algo completamente diferente.
Centrándonos más en $\mathbb{C}^n$, el espacio vectorial complejo cuyo elementos son n-tuplas $(z_1, z_2, \dots z_n)$, usaremos a veces la notación de vector columna:

$$\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{bmatrix}$$

2.2. Bases y dimensión

Definición 2.2. *Sea $|v_1\rangle, |v_2\rangle, \dots, |v_n\rangle$ vectores de un cierto espacio vectorial $\mathbb{V}$ sobre $\mathbb{C}$. Diremos que un vector $|v\rangle \in \mathbb{V}$ es **combinación lineal** de ellos si existen $a_1, a_2, \dots, a_n \in \mathbb{C}$ escalares tal que podemos escribir $|v\rangle$ como:*

$$|v\rangle = \sum_{i=1}^{n} a_i \cdot |v_i\rangle \tag{2.5}$$

Definición 2.3. *Sea $\{|v_1\rangle, |v_2\rangle, \ldots, |v_n\rangle\}$ un conjunto de vectores de un cierto espacio vectorial $\mathbb{V}$ sobre $\mathbb{C}$. Diremos que son* ***linealmente dependientes*** *si existen $a_1, a_2, \ldots, a_n \in \mathbb{C}$, con algún $a_i \neq 0$, tal que*

$$a_1 |v_1\rangle + a_2 |v_2\rangle + \ldots + a_n |v_n\rangle = 0 \tag{2.6}$$

Además diremos que son ***linealmente independientes*** *si no son linealmente dependientes. Es decir, si existe una combinación lineal de ellos, entonces los coeficientes son todos nulos.*

Definición 2.4. *Llamaremos entonces al conjunto $B = \{|v_1\rangle, |v_2\rangle, \ldots, |v_n\rangle\}$* ***base*** *del espacio $\mathbb{V}$ si:*

- *B es linealmente independiente.*
- *$\forall |v\rangle \in \mathbb{V}$, $|v\rangle$ puede ser escrito como combinación lineal de vectores de B.*

Además podemos asegurar la existencia de este conjunto para todo espacio vectorial. Y también de que el número de elementos de dos bases distintas del mismo espacio vectorial coincide y nos referiremos a este número como **dimensión** del espacio $\mathbb{V}$.
Como hemos hecho mención antes, nuestro interés se halla en espacios vectoriales de dimensión finita, por tanto haremos omiso de las cuestiones relacionadas con espacios de dimensión infinita.

2.3. Aplicaciones lineales y forma matricial

Definición 2.5. *Una aplicación lineal entre dos espacios vectoriales $\mathbb{V}$ y $\mathbb{W}$ sobre el mismo cuerpo $\mathbb{C}$ es una aplicación $f : \mathbb{V} \longrightarrow \mathbb{W}$ tal que es lineal sobre sus componentes, es decir, si $|v\rangle = \sum_{i=1}^{n} a_i \cdot |v_i\rangle$ entonces se cumple:*

$$f(|v\rangle) = f(\sum_{i=1}^{n} a_i \cdot |v_i\rangle) = \sum_{i=1}^{n} a_i \cdot f(|v_i\rangle) \tag{2.7}$$

Diremos además que una aplicación lineal está definida sobre $\mathbb{V}$ para referirnos a que es una aplicación lineal de $\mathbb{V}$ a $\mathbb{V}$

Un aplicación de gran importancia es la aplicación identidad, que denotaremos con $id_{\mathbb{V}}$ y cumple la propiedad de que $\forall |v\rangle \in \mathbb{V}$, $id_{\mathbb{V}}(|v\rangle) = |v\rangle$.
Observando la expresión 2.7 podemos llegar a la conclusión de que una aplicación lineal está completamente determinada por su acción sobre los elementos de una base, pues todo vector se puede expresar como combinación lineal de los vectores de una base.
Una manera muy útil de expresar una aplicación lineal es a través de su expresión matricial. Veamos esto con la aplicación de $f : \mathbb{V} \longrightarrow \mathbb{W}$ sobre los vectores de las bases correspondientes. Sea $\{|v_1\rangle, \ldots, |v_m\rangle\}$ y $\{|w_1\rangle, \ldots, |w_n\rangle\}$ bases correspondientes a $\mathbb{V}$ y $\mathbb{W}$.
Entonces para cada j de 1 a m existirán $a_{1j}, \ldots, a_{nj} \in \mathbb{C}$ tal que

$$f(|v_j\rangle) = \sum_{i=1}^{n} a_{ij} |w_i\rangle \tag{2.8}$$

por ser $f(|v_j\rangle) \in \mathbb{W}$ y $\{|w_1\rangle, \ldots, |w_n\rangle\}$ base de $\mathbb{W}$.

Definición 2.6. *Llamaremos entonces A a la matriz formada por los elementos a_{ij} de la ecuación 2.8 en la posición (ij) en la matriz, como representación matricial de la función f.*

Además, tomando las **coordenadas** $\boldsymbol{z_j}$ de un vector $|v\rangle = \sum_{j=1}^{m} \boldsymbol{z_j} |v_j\rangle$ de $\mathbb{V}$ y su imagen por f con la expresión 2.8:

$$f(|v\rangle) = f(\sum_{j=1}^{m} z_j |v_j\rangle) = \sum_{j=1}^{m} z_j f(|v_j\rangle) = \sum_{j=1}^{m} z_j (\sum_{i=1}^{n} a_{ij} |w_i\rangle) = \sum_{i=1}^{n} (\sum_{j=1}^{m} a_{ij} z_j) |w_i\rangle \tag{2.9}$$

podemos observar la aplicación de f sobre el vector $|v\rangle$ no es más que el producto de la matriz A con el vector $|v\rangle$ en columnas:

$$A |v\rangle = \begin{bmatrix} a_{11} & a_{12} & \ldots & a_{1m} \\ a_{21} & a_{22} & \ldots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \ldots & a_{nm} \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_m \end{bmatrix} = \begin{bmatrix} \sum_{j=1}^{m} a_{1j} z_j \\ \sum_{j=1}^{m} a_{2j} z_j \\ \vdots \\ \sum_{j=1}^{m} a_{nj} z_j \end{bmatrix} \tag{2.10}$$

2.4. Producto Escalar y Espacios de Hilbert

Definición 2.7. *Un **producto escalar**, o también conocido como producto interno, es una aplicación* $(\cdot,\cdot) : \mathbb{V} \times \mathbb{V} \longrightarrow \mathbb{C}$ *que cumple:*

- *Es definida positiva,*

$$(|v\rangle, |v\rangle) \geq 0 \tag{2.11}$$

y

$$(|v\rangle, |v\rangle) = 0 \Leftrightarrow |v\rangle = 0 \tag{2.12}$$

- *Es lineal en el primer argumento y lineal conjugada en el segundo,*

$$(a|u\rangle + b|v\rangle, |w\rangle) = \bar{a} \cdot (|u\rangle, |w\rangle) + \bar{b} \cdot (|v\rangle, |w\rangle) \tag{2.13}$$

$$(|u\rangle, a|v\rangle + b|w\rangle) = a \cdot (|u\rangle, |v\rangle) + b \cdot (|u\rangle, |w\rangle) \tag{2.14}$$

- *Es anti-simétrico,*

$$(|u\rangle, |v\rangle) = \overline{(|v\rangle, |u\rangle)} \tag{2.15}$$

donde $a, b \in \mathbb{C}$, $|u\rangle, |v\rangle, |w\rangle \in \mathbb{V}$ *y* $\bar{a} \in \mathbb{C}$ *es el conjugado complejo del elemento* a.

La notación estándar del producto escalar en mecánica cuántica no es $(|u\rangle, |v\rangle)$, sino $\langle u|v\rangle$, donde $|u\rangle$ y $|v\rangle$ son vectores de $\mathbb{V}$ y $\langle u|$ denota el vector dual al vector $|u\rangle$, también conocido como **bra**. El dual es una aplicación lineal cuya definición es $\langle u|(|v\rangle) := \langle u|v\rangle = (|u\rangle, |v\rangle)$. A partir de ahora, usaremos más esta notación.

Definición 2.8. *Diremos que dos vectores* $|u\rangle$ *y* $|v\rangle$ *son **ortogonales** si su producto escalar es 0. Además definiremos como **norma** del vector como*

$$\||v\rangle\| = \sqrt{\langle v|v\rangle} \tag{2.16}$$

y diremos que $|v\rangle$ *es unitario o normalizado si* $\||v\rangle\| = 1$.

Definición 2.9. *Por tanto diremos que un conjunto,* $|i\rangle \in \mathbb{V}$ *de vectores es **ortonormal** si son vectores unitarios y además son ortogonales entre sí. Es decir,*

$$\forall |i\rangle, |j\rangle \in \mathbb{V} \ \langle i|j\rangle = \delta_{ij} = \begin{cases} 1 & si \quad i = j \\ 0 & si \quad i \neq j \end{cases} \tag{2.17}$$

Un espacio vectorial euclídeo no es más que un espacio vectorial dotado de un producto escalar. Trabajaremos a partir de ahora en un espacio vectorial complejo de dimensión finita y con un producto escalar. Dicho espacio es denominado usualmente como **espacio de Hilbert**.

En nuestro caso por la finitud de la dimensión, un espacio de Hilbert es equivalente a espacio euclídeo. No entraremos en detalles en el caso de que la dimensión sea infinita, ya que para hablar de espacio de Hilbert sería necesario que se cumplan alguna propiedad extra. Nos centraremos en el caso de la dimensión finita cuando hablemos de espacio de Hilbert.

Podemos ver ahora que el producto escalar en un espacio de Hilbert tiene una representación matricial muy útil. Consideramos $|u\rangle = \sum_i u_i |i\rangle$ y $|v\rangle = \sum_j v_j |j\rangle$ con $|i\rangle, |j\rangle$ vectores de una base ortonormal $\{|1\rangle, |2\rangle, \ldots, |n\rangle\}$. Entonces el producto escalar,

$$\langle u|v\rangle = (\sum_i u_i |i\rangle, \sum_j v_j |j\rangle) = \sum_{ij} \bar{u_i} v_j \langle i|j\rangle = \sum_{ij} \bar{u_i} v_j \delta_{ij} = \sum_i \bar{u_i} v_i \tag{2.18}$$

que claramente es el producto entre un vector fila conjugado y uno columna,

$$\langle u|v\rangle = [\bar{u_1}\bar{u_2}\ldots\bar{u_n}] \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} = \sum_i \bar{u_i} v_i \tag{2.19}$$

Podemos observar también que el vector dual $\langle u|$ se puede expresar como un vector fila cuyas componentes están conjugadas.
Una manera útil de ver las aplicaciones lineales es a través de su representación como **producto exterior**.

Definición 2.10. *Llamaremos producto exterior a la aplicación* $|u\rangle\langle v| : \mathbb{V} \longrightarrow \mathbb{W}$, *donde* $|v\rangle \in \mathbb{V}$ *y* $|u\rangle \in \mathbb{W}$,

$$|u\rangle\langle v|\,(|v'\rangle) = |u\rangle\langle v|v'\rangle = \langle v|v'\rangle \cdot |u\rangle \tag{2.20}$$

Considerando ahora una base ortonormal $\{|i\rangle\}_{1\leq i\leq n}$, podemos deducir la propiedad de completitud del producto exterior. Sea el vector $|v\rangle = \sum_i v_i\,|i\rangle$, teniendo en cuenta que $\langle i|v\rangle = v_i$, tenemos que la aplicación de $\sum_i |i\rangle\langle i|$ sobre el vector

$$(\sum_i |i\rangle\langle i|)(|v\rangle) = \sum_i |i\rangle\langle i|v\rangle = \sum_i v_i\,|i\rangle = |v\rangle \tag{2.21}$$

Lo que nos permite llegar a la conclusión de que $\sum_i |i\rangle\langle i|$ es equivalente a la identidad.
Teniendo en cuenta esta propiedad podemos conseguir la expresión de un aplicación lineal $f : \mathbb{V} \longrightarrow \mathbb{W}$, considerando $|v_i\rangle$ y $|w_j\rangle$ un base ortonormal de ambos espacios. Con la propiedad de completitud tenemos que

$$f \equiv id_{\mathbb{W}} \circ f \circ id_{\mathbb{V}} \equiv \sum_{ij}(|w_j\rangle\langle w_j|) \circ f \circ (|v_i\rangle\langle v_i|) \equiv \sum_{ij} \langle w_j|f(v_i)\rangle\,|w_j\rangle\langle v_i| \tag{2.22}$$

donde podemos concluir que el valor $\langle w_j|f(v_i)\rangle$ es el elemento de la columna i y fila j de la representación matricial de f en las bases correspondientes.
Además observamos que esto concuerda con la expresión de un vector y su dual como vector fila y columna pues el producto resultante de

$$\begin{bmatrix} w_1 \\ \vdots \\ w_n \end{bmatrix} [v_1 \dots v_n] = \begin{bmatrix} w_1v_1 & \dots & w_1v_n \\ \vdots & \ddots & \vdots \\ w_nv_1 & \dots & w_nv_n \end{bmatrix} \tag{2.23}$$

es una matriz, correspondiente a la aplicación lineal.

2.5. Matrices Adjuntas o Hermitianas

Veremos ahora un tipo de matriz y su función asociada que se comporta de una manera muy buena con el espacio de Hilbert.

Definición 2.11. *Consideramos una matriz* $A \in \mathbb{C}^{n\times n}$, *definiremos su* ***adjunta*** *o* ***conjugada Hermitiana*** *como la matriz traspuesta con los elementos conjugados y lo denotaremos como* $A^\dagger = \overline{A^T}$
Además diremos que A *es* ***hermintiana*** *si* $A^\dagger = A$ *y llamaremos a la aplicación lineal asociada, aplicación* ***auto-adjunta****.*

Podemos ver fácilmente que este tipo de matrices cumplen ciertas propiedades,

- $\forall\,|u\rangle, |v\rangle \in \mathbb{V}$
$$(|u\rangle, A\,|w\rangle) = (A^\dagger\,|u\rangle, |w\rangle) \tag{2.24}$$
- Definiremos por convenio la adjunta de un vector $|v\rangle^\dagger = \langle v|$, que concuerda con toda la notación que hemos estado usando. De esta manera, teniendo en cuenta que $(AB)^\dagger = B^\dagger A^\dagger$, tenemos que
$$(A\,|v\rangle)^\dagger = \langle v|\,A^\dagger \tag{2.25}$$

Otras matrices que nos interesan son las matrices **unitarias**. Son un tipo de matrices invertibles que cumplen que

$$U * U^\dagger = U^\dagger * U = I_n \tag{2.26}$$

2.6. Producto Tensorial

En esta sección estudiaremos el producto tensorial entre espacios vectoriales, una herramienta esencial para trabajar con sistemas cuánticos de varios elementos en esta área. Hablaremos de estos sistemas en secciones posteriores, por ahora nos centraremos en el producto tensorial.

Definición 2.12. *Consideramos $\mathbb{V}$ y $\mathbb{W}$ dos espacios vectoriales, llamaremos* ***producto escalar*** *a la aplicación bilineal $\otimes : \mathbb{V} \times \mathbb{W} \longrightarrow \mathbb{V} \otimes \mathbb{W}$, que lleva $|v\rangle \in \mathbb{V}$ y $|w\rangle \in \mathbb{W}$ a un elemento de $\mathbb{V} \otimes \mathbb{W}$ que llamaremos* ***tensor*** *y lo denotaremos por $|v\rangle \otimes |w\rangle$, o de manera abreviada $|v\rangle |w\rangle$, $|vw\rangle$.*

Además esta aplicación cumple las siguientes propiedades:

- Sea z un escalar y $|v\rangle$ y $|w\rangle$ elementos de $\mathbb{V}$ y $\mathbb{W}$ respectivamente,

$$z(|v\rangle \otimes |w\rangle) = (z\,|v\rangle) \otimes |w\rangle = |v\rangle \otimes (z\,|w\rangle) \tag{2.27}$$

- Para $|v_1\rangle$, $|v_2\rangle \in \mathbb{V}$ y $|w\rangle \in \mathbb{W}$ se tiene,

$$(|v_1\rangle + |v_2\rangle) \otimes |w\rangle = |v_1\rangle \otimes |w\rangle + |v_2\rangle \otimes |w\rangle \tag{2.28}$$

- Para $|v\rangle \in \mathbb{V}$ y $|w_1\rangle$, $|w_2\rangle \in \mathbb{W}$ se tiene,

$$|v\rangle \otimes (|w_1\rangle + |w_2\rangle) = |v\rangle \otimes |w_1\rangle + |v\rangle \otimes |w_2\rangle \tag{2.29}$$

El espacio de la imagen sigue siendo un espacio vectorial y de hecho, si tomamos $\{|v_1\rangle, \ldots, |v_m\rangle\}$ y $\{|w_1\rangle, \ldots, |w_n\rangle\}$ como bases de $\mathbb{V}$ y $\mathbb{W}$ respectivamente, tenemos que

$$\{|v_i\rangle \otimes |w_j\rangle \mid 1 \leq i \leq m, 1 \leq j \leq n\} \tag{2.30}$$

es una base de $\mathbb{V} \otimes \mathbb{W}$. Y por tanto, la dimensión como espacio vectorial de $\mathbb{V} \otimes \mathbb{W}$ es $m \cdot n$ siendo m y n la dimensión de $\mathbb{V}$ y $\mathbb{W}$ respectivamente.
Las aplicaciones lineales del espacio $\mathbb{V} \otimes \mathbb{W}$ que consideraremos serán aquellas resultantes del producto tensorial de dos aplicaciones lineales del espacio de los factores, de manera que cumplan

$$(f \otimes g)(|v\rangle \otimes |w\rangle) = f(|v\rangle) \otimes g(|w\rangle). \tag{2.31}$$

De hecho, toda aplicación lineal de $\mathbb{V} \otimes \mathbb{W}$ se puede representar como combinación lineal de aplicaciones de $\mathbb{V}$ y $\mathbb{W}$ con el producto tensorial, actuando como espacio resultante del producto tensorial del espacio de los endomorfismos.
En cuanto a la práctica resulta muy cómodo trabajar con la representación matricial de estas aplicaciones y el producto de Kronecker. Pues si consideramos A una matriz $m \times n$ y B una matriz $p \times q$, su producto tensorial sería:

$$A \otimes B = \begin{bmatrix} a_{11}B & a_{12}B & \ldots & a_{1n}B \\ a_{21}B & a_{22}B & \ldots & a_{2n}B \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1}B & a_{m2}B & \ldots & a_{mn}B \end{bmatrix} \tag{2.32}$$

donde a_{ij} es el elemento de la posición (ij) de la matriz A.
De la misma manera se puede operar con los vectores columnas del espacio vectorial $\mathbb{C}^n$. Por ejemplo, si tenemos $|v\rangle\, y\, |w\rangle$ vectores de $\mathbb{C}^n$ y $v_1, v_2, \ldots, v_n$, $w_1, w_2, \ldots, w_n$ sus coordenadas respectivamente en

una base. Entonces su producto tensorial en forma matricial sería,

$$\begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} \otimes \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix} = \begin{bmatrix} v_1 \cdot \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix} \\ v_2 \cdot \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix} \\ \vdots \\ v_n \cdot \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix} \end{bmatrix} = \begin{bmatrix} v_1 w_1 \\ v_1 w_2 \\ \vdots \\ v_1 w_n \\ v_2 w_1 \\ v_2 w_2 \\ \vdots \\ v_2 w_n \\ v_n w_1 \\ v_n w_2 \\ \vdots \\ v_n w_n \end{bmatrix} \tag{2.33}$$

3. Estructuras y Puertas Cuánticas

En esta sección veremos el sistema de información en el que se basa la computación cuántica y su representación matemática. Igual que en la computación clásica se basa en el concepto de bit, en la computación cuántica se estudiará el **bit cuántico** o **qubit** (de quantum bit en inglés). Es verdad que el qubit, al igual que el bit, son objetos físicos de un sistema físico real como partículas subatómicas en un ordenador cuántico. Pero nosotros nos centraremos en describir el qubit como un objeto matemático abstracto con ciertas propiedades determinadas.

3.1. Qubit

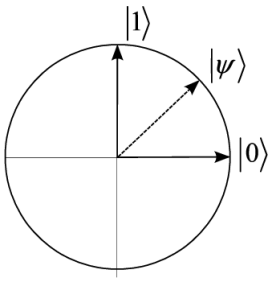
Conocemos el concepto de bit, que es un elemento de un sistema con dos estados posibles. Estos dos estados son generalmente denominados como "verdadero.º "falso, o incluso con 0 o 1.

Definición 3.1. *Entonces llamaremos **qubit** al objeto matemático con dos posibles estados correspondientes al bit clásico $|0\rangle$ y $|1\rangle$, además de una combinación lineal de estos dos estados que llamaremos como **superposición**:*

$$|\varphi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle \tag{3.1}$$

Con α y β números complejos que cumplen que $|\alpha|^2 + |\beta|^2 = 1$.

Se entiende fácilmente el estado de un qubit como un vector del espacio vectorial complejo de dimensión 2, restringido a la circunferencia unidad, donde los estados $|0\rangle$ y $|1\rangle$ son los elementos de la base ortonormal del espacio con α y β como coordenadas del vector de norma 1. Podemos determinar



el estado de un bit clásico a la hora de examinarlo, pero en el caso del qubit, no podemos determinar el estado cuántico de un qubit en superposición, es decir, no podemos hallar el valor de α y β. Lo que sí podemos hacer es **medir** el qubit, determinar si colapsa en el estado $|0\rangle$ con probabilidad $|\alpha|^2$ ó en el estado $|1\rangle$ con probabilidad $|\beta|^2$. En otra palabras, el proceso de medir un qubit, nos devuelve como salida un estado clásico al que colapsa y deja de estar en superposición de varios estados de manera simultanea.

Veamos ahora otra representación geométrica del qubit que puede resultar útil. Consideramos un qubit en estado de superposición.

$$|\psi\rangle = c_0\,|0\rangle + c_1\,|1\rangle \tag{3.2}$$

con $\|c_0|^2 + |c_1|^2 = 1$. Reescribimos la expresión en coordenadas en forma exponencial de un número complejo,

$$|\psi\rangle = r_0 e^{i\varphi_0}\,|0\rangle + r_1 e^{i\varphi_1}\,|1\rangle \tag{3.3}$$

Lo multiplicamos por un escalar de módulo 1, así no alteramos su estado cuántico,

$$e^{-i\varphi_0}\,|\psi\rangle = e^{-i\varphi_0}(r_0 e^{i\varphi_0}\,|0\rangle + r_1 e^{i\varphi_1}\,|1\rangle) = r_0\,|0\rangle + r_1 e^{i(\varphi_1-\varphi_0)}\,|1\rangle \tag{3.4}$$

Tomando ahora coordenadas polares $r_0 = \cos(\theta)$, $r_1 = \sin(\theta)$ y el cambio de variables $\varphi = \varphi_1 - \varphi_0$, resulta en la siguiente expresión,

$$|\psi\rangle = \cos(\theta)\,|0\rangle + e^{i\varphi}\sin(\theta)\,|1\rangle \tag{3.5}$$

Considerando θ y φ como coordenadas de un punto en una esfera tridimensional, podemos ver el estado del qubit como un punto en la superficie de dicha esfera, siendo los polos los estados $|0\rangle$ y $|1\rangle$. Esta esfera recibe el nombre de **esfera de Bloch** y veremos en poco que con ella podemos representar operaciones de qubits como rotaciones de la esfera.

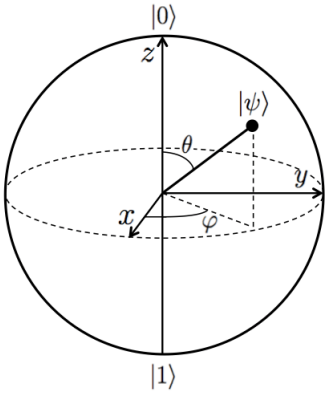


Figura 1: Esfera de Bloch.

3.2. Sistema de Varios Qubits

Tomamos interés en un sistema de un número mayor de qubits, que enlazaremos a través del producto tensorial de los espacios vectoriales. Por tanto un elemento de él estará representado por un vector de $\mathbb{C}^2 \otimes \mathbb{C}^2 \otimes \ldots \otimes \mathbb{C}^2$ que denotaremos como $(\mathbb{C}^2)^{\otimes n}$ donde n es el número de qubits del sistema y cuya dimensión como espacio vectorial es 2^n.

Por ejemplo, si queremos un sistema de 2 qubits en comparación a los estados clásicos de un bit tendríamos 00, 01, 10 y 11, por tanto nuestro sistema de 2 qubits tendría los estados $|00\rangle$, $|01\rangle$, $|10\rangle$ y $|11\rangle$ que se corresponden con el producto tensorial del estado de los dos qubits $|x\rangle \otimes |y\rangle$ con $x, y \in \{0, 1\}$. Vamos a trabajar más cómodamente con coordenadas del vector en su expresión matricial, consideramos primero,

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \ |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{3.6}$$

Como hemos visto en la sección anterior, aplicaremos el producto de Kroner para obtener las expresiones matriciales de $|00\rangle$, $|01\rangle$, $|10\rangle$ y $|11\rangle$,

$$|00\rangle = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \ |01\rangle = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \ |10\rangle = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \ |11\rangle = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \tag{3.7}$$

Observamos que forman una base ortonormal de un espacio vectorial de dimensión $2^2 = 4$ y que además cada coordenada corresponde a un estado clásico del qubit:

$$|01\rangle = \begin{matrix} \mathbf{00} \\ \mathbf{01} \\ \mathbf{10} \\ \mathbf{11} \end{matrix} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \tag{3.8}$$

Por tanto, si tomamos un sistema de dos qubits arbitrario en superposición,

$$|\psi\rangle = c_0 |00\rangle + c_1 |01\rangle + c_2 |10\rangle + c_3 |11\rangle \tag{3.9}$$

su representación matricial sería:

$$|\psi\rangle = \begin{matrix} \mathbf{00} \\ \mathbf{01} \\ \mathbf{10} \\ \mathbf{11} \end{matrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} \tag{3.10}$$

Trabajaremos a partir de ahora con la notación del sistema de varios qubits como $|xy\rangle$, $|x \otimes y\rangle$, ó $|x\rangle \otimes |y\rangle$ y los trataremos de manera indiferente según convenga.

Podemos medir únicamente un subconjunto de qubits del sistema total siendo la probabilidad de obtener el qubit en un estado concreto la suma de las probabilidades del estado del sistema con el qubit correspondiente en el estado en concreto. Por ejemplo si tomamos el sistema de dos qubits anterior, la probabilidad de que el primer qubit esté en el estado $|0\rangle$ será $|c_0|^2 + |c_1|^2$ que son las probabilidades de que el sistema esté en el estado $|00\rangle$ ó $|01\rangle$. Hacemos notar que tras medirlo, el sistema quedará en el estado

$$|\psi'\rangle = \frac{c_0 |00\rangle + c_1 |01\rangle}{\sqrt{|c_0|^2 + |c_1|^2}} \tag{3.11}$$

ya que hemos determinado el estado del primer qubit haciéndolo colapsar a un estado clásico a la hora de medirlo. Además, el denominador $\sqrt{|c_0|^2 + |c_1|^2}$ se debe por normalizar el vector que representa el estado del sistema. Recordemos que esto era una condición necesaria que tenemos que pedir. Un ejemplo importante de un sistema de dos qubits es el **estado de Bell** ó **par ERP**,

$$\frac{|00\rangle + |11\rangle}{\sqrt{2}} \tag{3.12}$$

que nos muestra una propiedad importante de los sistemas de varios qubits a la hora de hacer mediciones. Si medimos el primer qubit tendrá una probabilidad de $(\frac{1}{\sqrt{2}})^2 = \frac{1}{2}$ de estar en el estado $|0\rangle$ y una probabilidad de $\frac{1}{2}$ de estar en el estado $|1\rangle$. Esto dejará el sistema en el estado $|\psi'\rangle = |00\rangle$ ó $|\psi'\rangle = |11\rangle$. En cualquier caso tenemos que ambas situaciones, tras medir el primer qubit, si medimos el segundo este coincidirá con el primero. Es decir, que ambos qubits está enlazados.

Nos fijamos un momento en la dimensión del espacio vectorial que representa nuestro sistema. Supongamos que tenemos un número de un tamaño considerable de qubits, $n = 500$. Si quisiéramos simular este sistema en un ordenador clásico realizando las operaciones matriciales necesarias en la computación, tendríamos que para un estado del sistema almacenar 2^{500} coordenadas complejas del vector además de trabajar con matrices de tamaño 2^{500}. Esto claramente no es factible hoy en día resultando en una de los problemas en cuanto a la computación cuántica.

3.3. Circuitos y Puertas Cuánticas

Al igual que un circuito de un ordenador clásico consiste en cables y puertas lógicas que interaccionan con la información que es transportada por el cableado, un circuito cuántico consiste en cables que transportan información cuántica en forma de qubits que son manipulados a través de puertas cuánticas.

Consideremos primero la puerta clásica **NOT** que queremos que realice la operación de la tabla de verdad sobre un bit, es decir,

$$\begin{cases} 0 \to 1 \\ 1 \to 0 \end{cases} \tag{3.13}$$

intercambiando los estados 0 y 1. Queremos que nuestra puerta cuántica se comporte de la misma manera sobre qubits. Queremos que intercambie los estados $|0\rangle$ y $|1\rangle$ y además por ser puerta cuántica pediremos que sea lineal, es decir, si tenemos un un qubit arbitrario en estado de superposición

$$\alpha |0\rangle + \beta |1\rangle \tag{3.14}$$

queremos que se aplique de forma lineal a cada uno de los términos de la suma,

$$\alpha |1\rangle + \beta |0\rangle . \tag{3.15}$$

Igual que hemos asociado con vectores a los bits, y podemos operar con los vectores de forma matricial, las puertas lógicas están asociados a las matrices. Por tanto si tomamos la matriz NOT $= \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$, esta matriz cumple que

$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \text{ y } \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{3.16}$$

tomando el vector correspondiente al estado cuántico $|0\rangle$ y $|1\rangle$ y los intercambia:

$$\text{NOT}\,|0\rangle = |1\rangle \text{ y NOT}\,|1\rangle = |0\rangle \tag{3.17}$$

Otra propiedad que les pediremos a las puertas cuánticas es que sean reversibles. Esto es que dados la salida y la operación aplicada seamos capaces de determinar la entrada del circuito. Representaremos las puertas cuánticas con matrices unitarias que son reversibles por su propia adjunta. Recordemos que esto es:

$$U * U^\dagger = U^\dagger * U = I_n \tag{3.18}$$

Podemos poner de ejemplo, la operación AND, que toma dos bits y nos devuelve 1 si ambos inputs son 1. Esta operación no es reversible, ya que si conocemos solamente el output, no podemos determinar cuál ha sido el input de la operación.
De esta manera tenemos que, aparte de la propiedad de ser lineal que está implicada, cualquier matriz unitaria lleva asociada su puerta cuántica correspondiente, siendo el único requisito que pediremos. Veamos otras puertas cuánticas de un qubits de importancia. Si consideramos las matrices de Pauli,

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix},\ Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix},\ Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \tag{3.19}$$

tendremos que cada una de ellas tiene asociado una puerta cuántica correspondiente. De hecho, la matriz X corresponde con la puerta NOT. Podemos evaluar la matriz con los estados del qubit para determinar su comportamiento, pues por ejemplo si tomamos la matriz Z,

$$Z(\alpha\,|0\rangle + \beta\,|1\rangle) = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \alpha \\ -\beta \end{bmatrix} = \alpha\,|0\rangle - \beta\,|1\rangle\,, \tag{3.20}$$

que cambia el signo del coeficiente de $|1\rangle$.
Otra manera de visualizar estas puertas es como rotaciones de 180º sobre la esfera de Bloch en el eje correspondiente.

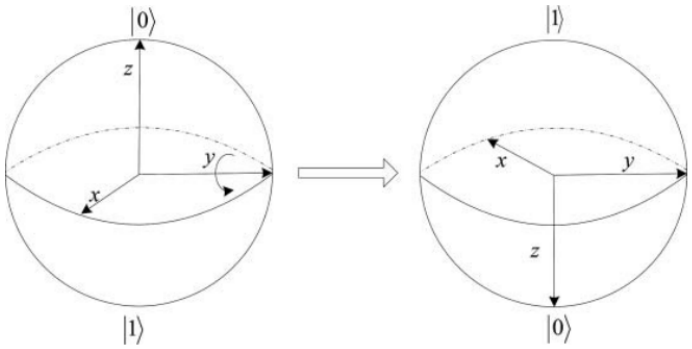


Figura 2: Rotación sobre el eje Y en la esfera de Bloch.

Una puerta cuántica de mucha importancia es la puerta de Hadamard ya que nos permitirá poner qubit en una configuración en la que todos los estados son equiprobables. Esta puerta tiene por matriz:

$$\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \tag{3.21}$$

que lleva

$$\begin{cases} |0\rangle \to \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \\ |1\rangle \to \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \end{cases} \tag{3.22}$$

Una propiedad que cumple $H^2 = I$ que se comprueba fácilmente. Esto nos permite posteriormente llevar estados $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ a $|0\rangle$, lo que usaremos posteriormente en los algoritmos.

Ya hemos mencionado que es posible visualizar las puertas cuánticas como rotaciones en la esfera de Bloch, por ejemplo si tomamos la expresión de un estado de un qubit en polares,

$$|\psi\rangle = \cos(\theta)\,|0\rangle + e^{i\varphi}\sin(\theta)\,|1\rangle\,, \tag{3.23}$$

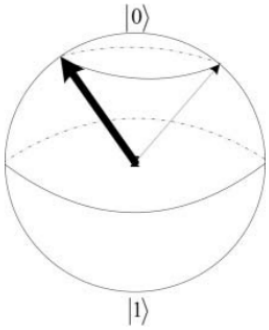
y le aplicamos la siguiente matriz

$$R(\omega) = \begin{bmatrix} 1 & 0 \\ 0 & e^{\omega} \end{bmatrix} \tag{3.24}$$

obteniendo como resultado:

$$|\psi'\rangle = \cos(\theta)\,|0\rangle + e^{\omega}e^{i\varphi}\sin(\theta)\,|1\rangle = \cos(\theta)\,|0\rangle + e^{i\varphi+\omega}\sin(\theta)\,|1\rangle \tag{3.25}$$

de esta manera hemos conseguido rotar la esfera alrededor del eje Z, por lo que no hemos variado lo que podemos considerar como latitud, sino hemos variado su longitud. Este tipo de operaciones se denominan **cambios de fase**, ya que sólo estamos alterando el valor del parámetro $e^{i\varphi}$, que denominaremos como **fase**. Observamos que esta alteración de la fase del estado del qubit no produce cambios en cuanto a la medición del mismo, ya que la probabilidad de que un qubit colapse en un estado clásico depende en este caso únicamente del parámetro θ, su latitud en la esfera de Bloch, o también **amplitud**.



Otras rotaciones, por ejemplos si queremos rotar la espera un cierto ángulo ω respecto a un eje, tendría la siguiente expresión:

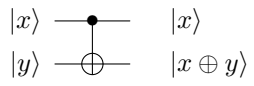
$$\begin{cases} R_x(\omega) = \cos(\frac{\omega}{2})I - i\sin(\frac{\omega}{2})X = \begin{bmatrix} cos(\frac{\omega}{2}) & -i\sin(\frac{\omega}{2}) \\ -i\sin(\frac{\omega}{2}) & \cos(\frac{\omega}{2}) \end{bmatrix} \\ R_y(\omega) = \cos(\frac{\omega}{2})I - i\sin(\frac{\omega}{2})Y = \begin{bmatrix} cos(\frac{\omega}{2}) & -\sin(\frac{\omega}{2}) \\ \sin(\frac{\omega}{2}) & \cos(\frac{\omega}{2}) \end{bmatrix} \\ R_z(\omega) = \cos(\frac{\omega}{2})I - i\sin(\frac{\omega}{2})Z = \begin{bmatrix} e^{-i\omega/2} & 0 \\ 0 & e^{i\omega/2} \end{bmatrix} \end{cases} \tag{3.26}$$

También podemos realizar rotaciones respecto de un vector dado, consideramos $D = (D_x, D_y, D_z)$ un vector arbitrario de módulo 1, con las coordenadas correspondientes. La matriz de la rotación estará determinada por la siguiente expresión:

$$R_D(\omega) = \cos(\frac{\omega}{2})I - i\sin(\frac{\omega}{2})(D_xX + D_yY + D_zZ) \tag{3.27}$$

Veamos ahora, puertas cuánticas que involucren más de un qubit, que a pesar de que no podremos representarlos ya en la esfera de Bloch, podemos seguir considerando su representación matricial.

Una de las puertas más importantes es la puerta **NOT controlada** o **CNOT**. Esta puerta tomará dos entradas y dará dos salidas. La primera entrada la llamaremos bit de control, es decir, controlará el bit de salida. Representaremos la puerta en la siguiente ilustración de un circuito cuántico, donde cada recta horizontal representa un cable que lleva el qubit siguiendo la lectura de izquierda a derecha. Si $|x\rangle = |0\rangle$, la salida del segundo bit $|y\rangle$ permanecerá igual. Si $|x\rangle = |1\rangle$ entonces $|y\rangle$ se le aplicará la



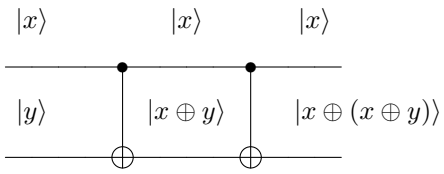
puerta NOT, es decir, será lo opuesto. Es decir, realiza la siguiente transformación

$$\begin{cases} |00\rangle \to |00\rangle \\ |01\rangle \to |01\rangle \\ |10\rangle \to |11\rangle \\ |11\rangle \to |10\rangle \end{cases} \tag{3.28}$$

Esto se puede ver como que la puerta transforma un par de bits $|x, y\rangle$ en $|x, x \oplus y\rangle$, donde $\oplus$ es la operación binaria de OR excluyente, o también como la suma en módulo 2. También podemos describir la puerta CNOT con su representación matricial, considerando como base del espacio vectorial $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$,

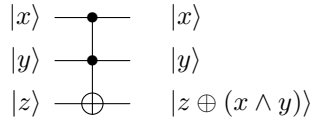
$$\text{CNOT} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{3.29}$$

Observamos que claramente esta puerta es unitaria pues el producto por su adjunta que es ella misma resulta en la matriz identidad, pues tenemos que se cumple que $CNOT^{\dagger}CNOT = I$. Esto también se puede entender como que la puerta CNOT es reversible por ella misma:



El estado de entrada es $|x, y\rangle$ que queda trasformado en $|x, x \oplus y\rangle$ y esta última en $|x, x \oplus (x \oplus y)\rangle$. Esto es $|x, (x \oplus x) \oplus y\rangle = |x, 0 \oplus y\rangle = |x, y\rangle$, ya que $x \oplus x = 0$.

Otra puerta reversible de mucha importancia es la de **Toffoli**, que funciona de manera similar a la puerta NOT controlada. Trabaja con 3 bits de entrada y salida, en el que aplica la puerta NOT al último bit $|z\rangle$ si y solo si los dos anteriores tienen por estado 1, $|x, y\rangle = |11\rangle$. En otras palabras lleva el $|x, y, z\rangle$ a $|x, y, z \oplus (x \wedge y)\rangle$.

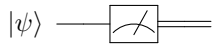


Esta puerta tiene por matriz:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{3.30}$$

Ya hemos visto varias puertas que son reversibles, ahora, llamaremos puertas cuánticas a aquellas aplicaciones que actúan sobre qubits, como vectores de un espacio vectorial complejo y tendrá como representación matricial, matrices unitarias.

Una excepción de las puertas cuánticas sería la operación de medir un qubit, que se suelen realizar al final del circuito y que no son reversibles. Representaremos estas de la siguiente manera en el circuito, donde la doble recta horizontal, representa un cable que lleva información clásica, un bit.



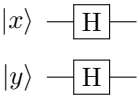
Hemos visto varios ejemplos de circuitos cuánticos con las puertas cuánticas. Vamos a centrarnos en construir circuitos más complejos combinando las puertas. Podemos tanto concatenar las puertas, realizando una operación tras otra, y operar de manera paralela con ellas, de manera simultanea a un subconjunto del sistema de qubits. Hacemos notar varias restricciones respecto al mundo clásico y es que no permitiremos bucles en los circuitos, de manera que cada circuito funcione de izquierda a derecha y sea acíclico. Además no permitiremos tanto combinar dos cableado en uno, como dividir uno en dos. Veremos en un ejemplo próximo que de hecho no podemos copiar el estado de un qubit a otro.

En cuanto a concatenar las puertas en un circuito, simplemente realizaremos una operación tras otra, de manera ordenada. Si queremos tratar la operación como una matriz, esta se representará por productos de matrices de manera natural. Por ejemplo si queremos aplicar dos veces la puerta CNOT como hemos hecho antes,

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.31}$$

resultando lo que esperábamos, ya que la puerta CNOT es reversible por sí misma, resultando que la concatenación de ella con sigo misma es la identidad.

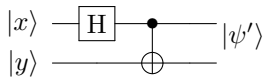
También podemos realizar operaciones de manera paralela a distintos bits a través de productos tensoriales entre las matrices, siempre cuidando el tamaño de las matrices. Por ejemplo, si queremos aplicar la puerta de Hadamard H a ambos qubits de un sistema de dos qubits, el resultado final será igual que aplicar $(H \otimes H)$ como una puerta al sistema, Recordemos la propiedad del producto tensorial sobre las



aplicaciones con la expresión 2.31, de esta manera tenemos que,

$$(H|x\rangle) \otimes (H|y\rangle) = (H \otimes H)|x \otimes y\rangle \tag{3.32}$$

Vamos a ver un ejemplo más de circuito en el que concatenemos la puerta CNOT con la puerta de Hadamard. Consideramos entonces el circuito:



Supongamos entonces que comenzaremos con el estado $|00\rangle$, para los otros casos el procedimiento es análogo. Entonces tendríamos que operar la siguiente expresión:

$$|\psi'\rangle = \text{CNOT} \cdot (H \otimes I)(|x \otimes y\rangle) \tag{3.33}$$

Hacemos notar que a pesar de que no le aplicamos ninguna puerta de manera simultanea a $|y\rangle$ para poder hacer el producto de matrices adecuadamente, es necesario aplicar el producto tensorial a la puerta de Hadamard con la identidad, que no altera el estado de $|y\rangle$.
Resolvamos entonces la expresión, con el estado $|00\rangle$ en concreto. Aplicando la propiedad de la expresión 3.32 , tenemos que:

$$|\psi'\rangle = \text{CNOT} \cdot (H \otimes I)(|00\rangle) = \text{CNOT}((H\,|0\rangle) \otimes (I\,|0\rangle)) = \text{CNOT}(\frac{|0\rangle + |1\rangle}{\sqrt{2}} \otimes |0\rangle) \tag{3.34}$$

Reordenando la expresión con las propiedades del producto escalar tenemos que:

$$|\psi'\rangle = \text{CNOT}(\frac{1}{\sqrt{2}}(|00\rangle + |10\rangle)) \tag{3.35}$$

y finamente, aplicando la linealidad de CNOT:

$$|\psi'\rangle = \frac{1}{\sqrt{2}}(\text{CNOT}\,|00\rangle + \text{CNOT}\,|10\rangle) = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \tag{3.36}$$

Si recordamos de la sección anterior, hemos producido con este circuito empezando en el estado $|00\rangle$ al estado de Bell que hemos visto.
Este desarrollo de naturaleza algebraica que hemos hecho, tiene su equivalente en operaciones matriciales, pues podríamos haber hecho el producto de las matrices:

$$\text{CNOT} \cdot (H \otimes I) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \left(\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right) \tag{3.37}$$

Si operamos, obtendremos la siguiente matriz:

$$\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & -1 \\ 1 & 0 & -1 & 0 \end{bmatrix} \tag{3.38}$$

Por tanto si operamos con $|00\rangle$, tendremos:

$$\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & -1 \\ 1 & 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} \tag{3.39}$$

que es la expresión matricial del estado de Bell

$$\frac{|00\rangle + |11\rangle}{\sqrt{2}} \tag{3.40}$$

4. Algoritmos Cuánticos

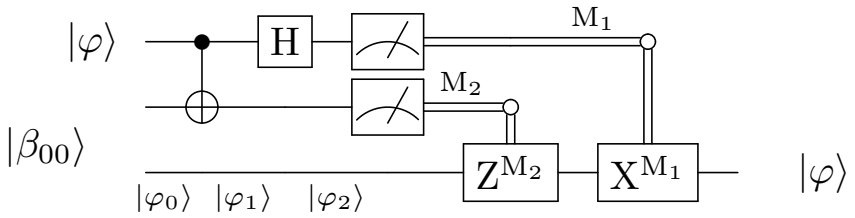
En general, los algoritmos cuánticos siguen un esquema común entre ellos. Estos consistirán en:

- Comenzaremos con una serie de qubits en su estado clásico, es decir, $|0\rangle$ ó $|1\rangle$
- Luego, serán puestos en superposición de varios estados.
- Se le aplicarán una serie de operaciones unitarias a través de puertas cuánticas.
- Y finalmente se medirán los qubits correspondientes y según el algoritmo, se repetirá este proceso varias veces y se compararán los resultados.

4.1. Algoritmo de Teletransportación

Empezaremos viendo un algoritmo sencillo que lo único que hará será teletransportar o mover un qubit en un estado arbitrario $|\varphi\rangle$ de un sitio a otro. Esta operación nos presenta los problemas más típicos del mundo cuántico, y es que no conocemos el estado $|\varphi\rangle$ en el que se encuentra el qubit, es decir, los valores de α y β de $|\varphi\rangle = \alpha|0\rangle + \beta|1\rangle$. Además de que no podemos determinar sus valores con sólo una copia del qubit, y que sólo sabemos enviar información de manera clásica en forma de bits.

El algoritmo toma ventaja del enlazamiento entre qubits para superar los problemas. Consideramos un transmisor y un receptor, además de un par de qubits que se encuentra en el estado de Bell. Nuestro transmisor tiene el control de uno de los qubits, mientras que el receptor tiene el control del otro. El algoritmo consistirá entonces en el transmisor interactua con el qubit $|\varphi\rangle$ y su correspondiente qubit del par en estado de Bell, seguido de tomar mediciones de los dos qubits y enviar la información de los bits resultantes al receptor. Este último aplicará ciertas operaciones dependiendo de la información recibida a su qubit del par en estado de Bell, resultado en el qubit $|\varphi\rangle$.
Veamos esto en el circuito y revisemos las cuentas paso a paso. Consideramos el estado inicial de entrada $|\varphi_0\rangle = |\varphi\rangle|\beta_{00}\rangle$ que es un sistema de tres qubits formados por el qubit $|\varphi\rangle$ y un par de qubits en estado de Bell.



Ponemos el estado inicial en la expresión que nos conviene usando la propiedad distributiva respecto de la suma.

$$\begin{aligned} |\varphi_0\rangle = |\varphi\rangle|\beta_{00}\rangle &= [\alpha|0\rangle + \beta|1\rangle] \otimes \left[\frac{|00\rangle + |11\rangle}{\sqrt{2}}\right] \\ &= \frac{1}{\sqrt{2}}[\alpha|0\rangle(|00\rangle + |11\rangle) + \beta|1\rangle(|00\rangle + |11\rangle)] \end{aligned} \tag{4.1}$$

Aplicamos la puerta CNOT a los dos primeros qubits, recordemos que la puerta CNOT cambia el estado del segundo qubit si el primero es un 1, mientras que si el primer qubit es un 0 dejará igual el sgundo qubit. Aprovechando entonces la linealidad de CNOT, tenemos por resultado

$$|\varphi_1\rangle = \frac{1}{\sqrt{2}}[\alpha|0\rangle(|00\rangle + |11\rangle) + \beta|1\rangle(|10\rangle + |01\rangle)] \tag{4.2}$$

Aplicamos entonces la puerta de Hadamard en el primer qubit, obteniendo

$$|\varphi_2\rangle = \frac{1}{\sqrt{2}}\left[\alpha\left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right)(|00\rangle + |11\rangle) + \beta\left(\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right)(|10\rangle + |01\rangle)\right] \tag{4.3}$$

Reordenamos la expresión

$$\begin{aligned} |\varphi_2\rangle = \frac{1}{2}\left[\alpha\left(|000\rangle + |011\rangle + |100\rangle + |111\rangle\right) + \beta\left(|010\rangle + |001\rangle - |110\rangle - |101\rangle\right)\right] = \\ \frac{1}{2}\left[|00\rangle\left(\alpha|0\rangle + \beta|1\rangle\right) + |01\rangle\left(\alpha|1\rangle + \beta|0\rangle\right) + |10\rangle\left(\alpha|0\rangle - \beta|1\rangle + |11\rangle\left(\alpha|1\rangle - \beta|0\rangle\right)\right)\right] \end{aligned} \tag{4.4}$$

Realizamos entonces la medición de los dos primeros qubits, que por estar enlazados los qubits, el resultado también determinará el estado del tercer qubit.

$$\begin{aligned} 00 &\longrightarrow \alpha|0\rangle + \beta|1\rangle \\ 01 &\longrightarrow \alpha|1\rangle + \beta|0\rangle \\ 10 &\longrightarrow \alpha|0\rangle - \beta|1\rangle \\ 11 &\longrightarrow \alpha|1\rangle - \beta|0\rangle \end{aligned} \tag{4.5}$$

Quedará finalmente realizar la transformación correspondiente al tercer qubit. Si el resultado de la medición es 00 entonces no es necesario realizar ningún cambio pues tenemos que el estado del tercer qubit es $\alpha|0\rangle + \beta|1\rangle = |\varphi\rangle$. Si es 01 tendremos que aplicar la puerta X para transformar el 1 en 0 y el 0 en 1. Si es 10 aplicaremos entonces la puerta Z que cambia el signo de la segunda coordenada, el coeficiente de $|1\rangle$. Y finalmente, en el caso de 11 aplicaremos primero la puerta X, seguido de la puerta Z. Esto sería equivalente a aplicar la transformación $Z^{M_1}X^{M_2}$ con M_1 y M_2 el valor de los dos primeros qubits, recuperando entonces el estado $|\varphi\rangle$ en el qubit que controla el receptor.

Hacemos notar que hemos llevado el estado $|\varphi\rangle$ de un qubit a otro, pero no lo hemos copiado, pues el estado del primer qubit ha sido sobreescrito con las operaciones realizadas. Como hemos mencionado antes, el proceso de copiar el estado de un qubit a otro no es posible. Podemos ver esto en que entra en contradicción con la propiedad lineal de las aplicaciones.
Suponemos C una aplicación correspondiente a la operación cuántica de copiar el estado de un qubit a otro. La hacemos actuar sobre un par de qubits resultando en

$$C\left(\frac{|x\rangle + |y\rangle}{\sqrt{2}} \otimes |0\rangle\right) = \left(\frac{|x\rangle + |y\rangle}{\sqrt{2}} \otimes \frac{|x\rangle + |y\rangle}{\sqrt{2}}\right) \tag{4.6}$$

Sin embargo, si aplicamos la propiedad lineal de C, tenemos que

$$\begin{aligned} C\left(\frac{|x\rangle + |y\rangle}{\sqrt{2}} \otimes |0\rangle\right) &= \frac{1}{\sqrt{2}}C\left(\left(|x\rangle + |y\rangle\right) \otimes |0\rangle\right) = \frac{1}{\sqrt{2}}C(|x\rangle \otimes |0\rangle + |y\rangle \otimes |0\rangle) \\ &= \frac{1}{\sqrt{2}}C(|x\rangle \otimes |0\rangle) + C(|y\rangle \otimes |0\rangle) = \frac{1}{\sqrt{2}}(|x\rangle \otimes |x\rangle) + (|y\rangle \otimes |y\rangle) \\ &= \frac{(|x\rangle \otimes |x\rangle) + (|y\rangle \otimes |y\rangle)}{\sqrt{2}} \end{aligned} \tag{4.7}$$

Que claramente llegamos a resultados distintos, concluyendo en que hemos encontrado una contradicción. Por tanto la aplicación C no es lineal y no se puede copiar estados de qubits.

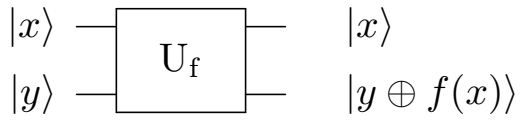
4.2. Algoritmo de Deutsch

Veamos ahora un algoritmo simple que llamaremos como el algoritmo de Deutsch. Este algoritmo tratará de ver si se cumple cierta propiedad para una función $f : \{0,1\} \to \{0,1\}$, concretamente si es balanceada o constante.

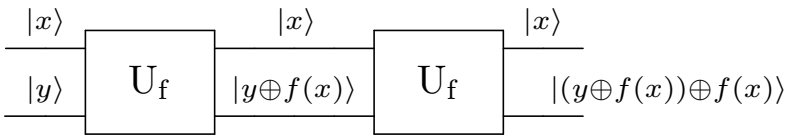
Definición 4.1. *Diremos que la función $f : \{0,1\} \to \{0,1\}$ es balanceada si $f(0) \neq f(1)$, y diremos que es constante si $f(0) = f(1)$.*

El algoritmo consistirá en tomar una función $f : \{0,1\} \to \{0,1\}$, que desconocemos su definición y sólo podremos evaluarla obteniendo su imagen con el objetivo de determinar si la función es balanceada o constante.
En un algoritmo clásico podríamos siempre evaluar la función en todos sus valores para determinar si

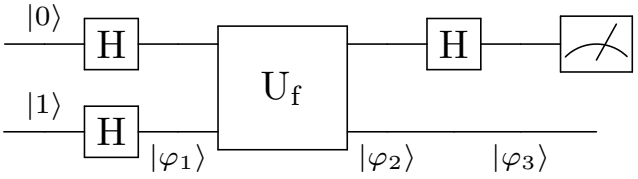
es balanceada o constate. Esto nos llevaría un total de dos evaluaciones de la función. Veamos si el algoritmo cuántico puede mejorar esto ya que nos permite evaluar de manera simultanea sobre todos los valores posibles.
Para ello tomaremos una puerta cuántica que evaluará la función y observamos que esta tiene que ser reversible. Consideraremos la siguiente operación unitaria U_f. A esta función a veces nos referiremos como oráculo, ya que nos evaluará la función indicando la imagen sin que nosotros sepamos la definición de la función. Recordemos también está representado por una matriz unitaria.



En la primera entrada $|x\rangle$ será lo que queremos evaluar, y en la segunda entrada $|y\rangle$ actuará como un qubit de control. Tras el cuál, en la primera salida $|x\rangle$ permanecerá igual y en la segunda salida tendremos $|y \oplus f(x)\rangle$.
Observamos que esta puerta cuántica es reversible pues:



El estado $|x, y\rangle$ queda trasformado en $|x, y \oplus f(x)\rangle$ y posteriormente en
$|x, (y \oplus f(x)) \oplus f(x)\rangle = |x, y \oplus (f(x) \oplus f(x))\rangle = |x, y \oplus 0\rangle = |x, y\rangle$, ya que $f(x) \oplus f(x) = 0$.
Una vez conseguido el oráculo que nos evaluará la función en los qubits, podemos construir el circuito del algoritmo que consistirá en lo siguiente:



Esto en términos matriciales sería:

$$(H \oplus I)U_f(H \oplus H)\,|0,1\rangle = (H \oplus I)U_f(H \oplus H) \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \tag{4.8}$$

Veamos en detalle el resultado al que se llega. Empezamos con el estado $|\varphi_0\rangle = |0,1\rangle$ que pondremos en superposición con la puerta de Hadamard. Hadamard transforma a $|0\rangle$ en $\frac{|0\rangle+|1\rangle}{\sqrt{2}}$ y $|1\rangle$ en $\frac{|0\rangle-|1\rangle}{\sqrt{2}}$, por tanto,

$$|\varphi_1\rangle = \left[\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right]\left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right] = \frac{|0,0\rangle - |0,1\rangle + |1,0\rangle - |1,1\rangle}{2} = \begin{bmatrix} +\frac{1}{2} \\ -\frac{1}{2} \\ +\frac{1}{2} \\ -\frac{1}{2} \end{bmatrix} \tag{4.9}$$

Ahora, aplicaremos el oráculo U_f a la expresión $\frac{|0,0\rangle-|0,1\rangle+|1,0\rangle-|1,1\rangle}{2}$ y recordando que U_f no deja de ser una aplicación lineal. Por tanto, tenemos que

$$|\varphi_2\rangle = U_f \frac{|0,0\rangle - |0,1\rangle + |1,0\rangle - |1,1\rangle}{2} = \frac{U_f\,|0,0\rangle - U_f\,|0,1\rangle + U_f\,|1,0\rangle - U_f\,|1,1\rangle}{2} \tag{4.10}$$

Recordemos que nuestro oráculo U_f transforma $|x,y\rangle$ en $|x, y \oplus f(x)\rangle$, por lo que

$$|\varphi_2\rangle = \frac{|0,0\oplus f(0)\rangle - |0,1\oplus f(0)\rangle + |1,0\oplus f(1)\rangle - |1,1\oplus f(1)\rangle}{2} = \frac{|0,f(0)\rangle - |0,\overline{f(0)}\rangle + |1,f(1)\rangle - |1,\overline{f(1)}\rangle}{2} \tag{4.11}$$

donde $\overline{f(x)}$ es el opuesto de $f(x)$.
Hacemos ver que $|x,y\rangle$ no es más que una notación de $|x\rangle \otimes |y\rangle$, por tanto aplicando la propiedad distributiva del producto tensorial por la derecha tenemos

$$|\varphi_2\rangle = \frac{|0\rangle \otimes \left[|f(0)\rangle - |\overline{f(0)}\rangle\right] + |1\rangle \otimes \left[|f(1)\rangle - |\overline{f(1)}\rangle\right]}{2} \tag{4.12}$$

Observemos en un momento la expresión $|f(x)\rangle - |\overline{f(x)}\rangle$ para $x \in \{0,1\}$ y discutimos su valor según el valor de $f(x)$.

$$|f(x)\rangle - |\overline{f(x)}\rangle = \begin{cases} |0\rangle - |1\rangle \text{ si } f(x)=0 \\ |1\rangle - |0\rangle \text{ si } f(x)=1 \end{cases} = \begin{cases} |0\rangle - |1\rangle \text{ si } f(x)=0 \\ -(|0\rangle - |1\rangle) \text{ si } f(x)=1 \end{cases} \tag{4.13}$$

que escribiremos como $(-1)^{f(x)}(|0\rangle - |1\rangle)$.
Aplicando esto a la expresión 4.12 tenemos que

$$|\varphi_2\rangle = \frac{(-1)^{f(0)}(|0\rangle \otimes [|0\rangle - |1\rangle]) + (-1)^{f(1)}(|1\rangle \otimes [|0\rangle - |1\rangle])}{2} \tag{4.14}$$

Hacemos uso de la propiedad distributiva por la derecha y reordenamos los escalares para separar la expresión en producto escalar de dos términos

$$|\varphi_2\rangle = \left[\frac{(-1)^{f(0)}\,|0\rangle + (-1)^{f(1)}\,|1\rangle}{\sqrt{2}}\right]\left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right] \tag{4.15}$$

Discutamos el valor de la expresión $(-1)^{f(0)}\,|0\rangle + (-1)^{f(1)}\,|1\rangle$ según si $f(x)$ es constante o balanceada.

$$|\varphi_2\rangle = \begin{cases} (\pm 1)\left[\frac{|0\rangle+|1\rangle}{2}\right]\left[\frac{|0\rangle-|1\rangle}{2}\right] & \text{si } f \text{ es constante,} \\ (\pm 1)\left[\frac{|0\rangle-|1\rangle}{2}\right]\left[\frac{|0\rangle-|1\rangle}{2}\right] & \text{si } f \text{ es balanceada.} \end{cases} \tag{4.16}$$

Teniendo en cuenta que Hadamard es su propia inversa, llevará $\frac{|0\rangle+|1\rangle}{2}$ a $|0\rangle$ y $\frac{|0\rangle-|1\rangle}{2}$ a $|1\rangle$.

$$|\varphi_3\rangle = \begin{cases} (\pm 1)\,|0\rangle\left[\frac{|0\rangle-|1\rangle}{2}\right] & \text{si } f \text{ es constante,} \\ (\pm 1)\,|1\rangle\left[\frac{|0\rangle-|1\rangle}{2}\right] & \text{si } f \text{ es balanceada.} \end{cases} \tag{4.17}$$

Finalmente medimos el qubit superior para determinar si f es constante o balanceada, ya que si sale $|0\rangle$ será constante y si sale $|1\rangle$ será balanceada. Observamos que el signo no afecta a la proceso de medir, pues recordemos que la probabilidad de que sea un estado depende de la norma al cuadrado.
Observamos que hemos conseguido en una evaluación de la función determinar si la propiedad de ser balanceada o constante. Sin embargo hacemos notar que para determinar cuál de los cuatro posibles funciones es, necesitamos evaluar de nuevo la función con este algoritmo.

4.3. Algoritmo de Deutsch-Jozsa

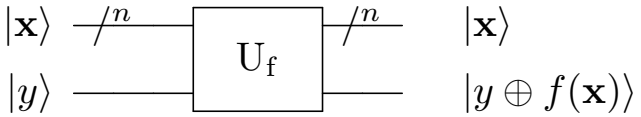
Veremos ahora el mismo problema pero generalizando la función. Consideramos ahora la función $f : \{0,1\}^n \longrightarrow \{0,1\}$ que lleva cadenas de n 0 y 1 a un único 0 ó 1 como imagen.

Definición 4.2. *Diremos entonces que $f : \{0,1\}^n \longrightarrow \{0,1\}$ es balanceada si exactamente la mitad de los elementos del dominio de la función tiene a 0 por imagen, y la otra mitad a 1. Diremos también que la función es constante si todos los elementos tiene como imagen el 0 ó si todos los elementos tiene como imagen el 1.*

El algoritmo de Deutsch-Jozsa resuelve el problema de dada una función $f : \{0,1\}^n \longrightarrow \{0,1\}$ con la que podemos evaluar pero no sabemos cómo está definido, suponiendo que la función sólo puede ser balanceada o constante, tenemos que determinar cuál de estas dos propiedades cumple.

Veamos primero como se podría resolver utilizando un método clásico. Se podría evaluar la función diferentes valores para poder determinar si es balanceada o constante. En el mejor de los casos, si tras dos evaluaciones se llega a dos imágenes distintas, podemos concluir que es balanceada pues está asegurado que cumpla una de las propiedades. En el peor de los casos se requiere evaluar la mitad de los elementos más uno, si todas las imágenes comprobadas son iguales se llega a que la función es constante, en otro caso será balanceada. Esto requiere en el pero de los casos $\frac{2^n}{2} + 1$ evaluaciones de la función, veamos si con el algoritmo cuántico se puede mejorar.

El algoritmo consistirá como el anterior, pondremos los qubits en estado de superposición para poder evaluarlos de manera simultanea en el oráculo. Este tendrá la siguiente forma, que se podrá representar con una matriz unitaria.



donde $\not{}^{n}$ indica que hay n qubits y usaremos $|\mathbf{x}\rangle = |x_0 x_1 \dots x_{n-1}\rangle$ para denotar la cadena binaria. En este caso, $|\mathbf{x}\rangle$ quedará igual tras el oráculo U_f, mientras que $|y\rangle$ será el qubit de control. Es fácil ver que, como en el algoritmo anterior, el oráculo U_f es reversible por sí misma.

Una vez obtenido el oráculo, queremos poner en superposición los n qubits. Para ello usaremos la matriz de Hadamard, concretamente el producto tensorial de n matrices de Hadamard. Veamos primero qué forma tiene esta matriz. Recordemos que la matriz de Hadamard tiene esta forma:

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{4.18}$$

Observamos que el elemento de la fila i columna j se puede expresar como $H[i,j] = \frac{1}{\sqrt{2}}(-1)^{i \wedge j}$, con i y j expresado en binario y $\wedge$ es la operación AND. De esta manera, la matriz de queda escrito como:

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} (-1)^{0\wedge 0} & (-1)^{0\wedge 1} \\ (-1)^{1\wedge 0} & (-1)^{1\wedge 1} \end{bmatrix} \tag{4.19}$$

Calculemos ahora el producto tensorial de dos matrices de Hadamard, que lo podemos expresar en forma de potencias de matrices con el producto tensorial que denotaremos como

$$H^{\otimes 2} = H \otimes H = \frac{1}{\sqrt{2}} \begin{bmatrix} (-1)^{0\wedge 0} & (-1)^{0\wedge 1} \\ (-1)^{1\wedge 0} & (-1)^{1\wedge 1} \end{bmatrix} \otimes \frac{1}{\sqrt{2}} \begin{bmatrix} (-1)^{0\wedge 0} & (-1)^{0\wedge 1} \\ (-1)^{1\wedge 0} & (-1)^{1\wedge 1} \end{bmatrix} \tag{4.20}$$

$$H^{\otimes 2} = \frac{1}{\sqrt{2}} \cdot \frac{1}{\sqrt{2}} \begin{bmatrix} (-1)^{0\wedge 0}\cdot(-1)^{0\wedge 0} & (-1)^{0\wedge 0}\cdot(-1)^{0\wedge 1} & (-1)^{0\wedge 1}\cdot(-1)^{0\wedge 0} & (-1)^{0\wedge 1}\cdot(-1)^{0\wedge 1} \\ (-1)^{0\wedge 0}\cdot(-1)^{1\wedge 0} & (-1)^{0\wedge 0}\cdot(-1)^{1\wedge 1} & (-1)^{0\wedge 1}\cdot(-1)^{1\wedge 0} & (-1)^{0\wedge 1}\cdot(-1)^{1\wedge 1} \\ (-1)^{1\wedge 0}\cdot(-1)^{0\wedge 0} & (-1)^{1\wedge 0}\cdot(-1)^{0\wedge 1} & (-1)^{1\wedge 1}\cdot(-1)^{0\wedge 0} & (-1)^{1\wedge 1}\cdot(-1)^{0\wedge 1} \\ (-1)^{1\wedge 0}\cdot(-1)^{1\wedge 0} & (-1)^{1\wedge 0}\cdot(-1)^{1\wedge 1} & (-1)^{1\wedge 1}\cdot(-1)^{1\wedge 0} & (-1)^{1\wedge 1}\cdot(-1)^{1\wedge 1} \end{bmatrix} \tag{4.21}$$

Teniendo en cuenta que estamos interesados sólo por el signo de $(-1)^x$ que depende de la paridad de x, podemos considerar el producto de $(-1)^x \cdot (-1)^y$ como $(-1)^{x\oplus y}$, donde $\oplus$ es la suma módulo 2. Entonces, la matriz sería

$$\begin{aligned} H^{\otimes 2} = \frac{1}{2} \begin{bmatrix} (-1)^{0\wedge 0\oplus 0\wedge 0} & (-1)^{0\wedge 0\oplus 0\wedge 1} & (-1)^{0\wedge 1\oplus 0\wedge 0} & (-1)^{0\wedge 1\oplus 0\wedge 1} \\ (-1)^{0\wedge 0\oplus 1\wedge 0} & (-1)^{0\wedge 0\oplus 1\wedge 1} & (-1)^{0\wedge 1\oplus 1\wedge 0} & (-1)^{0\wedge 1\oplus 1\wedge 1} \\ (-1)^{1\wedge 0\oplus 0\wedge 0} & (-1)^{1\wedge 0\oplus 0\wedge 1} & (-1)^{1\wedge 1\oplus 0\wedge 0} & (-1)^{1\wedge 1\oplus 0\wedge 1} \\ (-1)^{1\wedge 0\oplus 1\wedge 0} & (-1)^{1\wedge 0\oplus 1\wedge 1} & (-1)^{1\wedge 1\oplus 1\wedge 0} & (-1)^{1\wedge 1\oplus 1\wedge 1} \end{bmatrix} \\ = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \end{aligned} \tag{4.22}$$

De esta manera si definimos la aplicación $\langle\ ,\ \rangle : \{0,1\}^n \times \{0,1\}^n \longrightarrow \{0,1\}$ de manera que dados dos cadenas binarias de longitud n, $\mathbf{x} = x_0x_1\ldots x_{n-1}$ y $\mathbf{y} = y_0y_1\ldots y_{n-1}$, se define

$$\begin{aligned} \langle \mathbf{x}, \mathbf{y}\rangle &= \langle x_0x_1\ldots x_{n-1}, y_0y_1\ldots y_{n-1}\rangle \\ &= (x_0 \wedge y_0) \oplus (x_1 \wedge y_1) \oplus \ldots \oplus (x_{n-1} \wedge y_{n-1}) \end{aligned} \tag{4.23}$$

Observamos que esta aplicación es en realidad el producto escalar del espacio vectorial cuyo cuerpo es el espacio $\mathbb{Z}_2$. Por tanto, podemos expresar $H^{\otimes 2}$ como

$$\begin{bmatrix} (-1)^{\langle 00,00\rangle} & (-1)^{\langle 00,01\rangle} & (-1)^{\langle 00,10\rangle} & (-1)^{\langle 00,11\rangle} \\ (-1)^{\langle 01,00\rangle} & (-1)^{\langle 01,01\rangle} & (-1)^{\langle 01,10\rangle} & (-1)^{\langle 01,11\rangle} \\ (-1)^{\langle 10,00\rangle} & (-1)^{\langle 10,01\rangle} & (-1)^{\langle 10,10\rangle} & (-1)^{\langle 10,11\rangle} \\ (-1)^{\langle 11,00\rangle} & (-1)^{\langle 11,01\rangle} & (-1)^{\langle 11,10\rangle} & (-1)^{\langle 11,11\rangle} \end{bmatrix} \tag{4.24}$$

Ahora, si queremos generalizarlo para $H^{\otimes n}$ tenemos que su elemento de la fija i columna j expresado los indices en binario, es

$$H^{\otimes n} = \frac{1}{\sqrt{2^n}} (-1)^{\langle i,j\rangle} \tag{4.25}$$

Entonces, si consideramos un estado arbitrario $|\mathbf{y}\rangle$ que estará representado por un vector con un 1 en la posición $\mathbf{y}$ y el resto 0 y le aplicamos $H^{\otimes n}$, obtendremos la suma de los elementos de la columna $\mathbf{y}$ de la matriz, es decir,

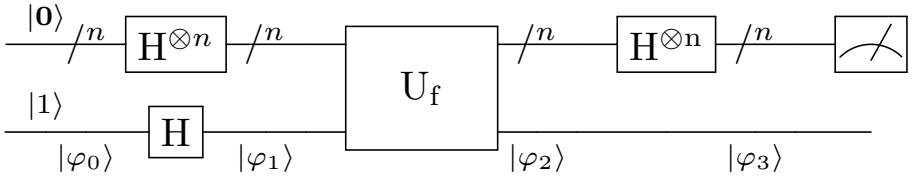
$$H^{\otimes n} |\mathbf{y}\rangle = \frac{1}{\sqrt{2^n}} \sum_{\mathbf{x}\in\{0,1\}^n} (-1)^{\langle \mathbf{x},\mathbf{y}\rangle} |\mathbf{x}\rangle \tag{4.26}$$

Así podremos poner nuestros qubits en un estado de superposición de todos los estados posibles. Por ejemplo podemos tomar el estado $|\mathbf{0}\rangle$ y al aplicarle $H^{\otimes n}$ esta será

$$H^{\otimes n} |\mathbf{0}\rangle = \frac{1}{\sqrt{2^n}} \sum_{\mathbf{x}\in\{0,1\}^n} (-1)^{\langle \mathbf{x},\mathbf{0}\rangle} |\mathbf{x}\rangle = \frac{1}{\sqrt{2^n}} \sum_{\mathbf{x}\in\{0,1\}^n} |\mathbf{x}\rangle \tag{4.27}$$

ya que $\langle \mathbf{x}, \mathbf{0}\rangle = 0$, para cualquier valor $\mathbf{x}$.

Una vez aclarado esto, podemos proceder a ver el circuito del algoritmo. De manera similar al algoritmo anterior, comenzaremos con $|\mathbf{0}\rangle = |00\ldots0\rangle$ siendo una cadena de 0 como el estado de los primeros n qubits, y $|1\rangle$ el qubit de control. Aplicaremos la puerta $H^{\otimes n}$ para poner los primeros n qubits en superposición.



O en término de matrices, esto sería:

$$\left(H^{\otimes n} \otimes I\right) U_f \left(H^{\otimes n} \otimes H\right) |\mathbf{0}, 1\rangle \tag{4.28}$$

Empezamos entonces con el estado $|\varphi_0\rangle = |\mathbf{0}, 1\rangle$ y le aplicamos la matriz de Hadamard, usando la expresión 4.27 y reordenando por la propiedad distributiva

$$\begin{aligned} |\varphi_1\rangle = \left(H^{\otimes n} \otimes H\right) |\mathbf{0}, 1\rangle = \left[H^{\otimes n} |\mathbf{0}\rangle\right] \otimes \left[H |1\rangle\right] = \left[\frac{1}{\sqrt{2^n}} \sum_{\mathbf{x}\in\{0,1\}^n} |\mathbf{x}\rangle\right] \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right] \\ = \frac{1}{\sqrt{2^n}} \sum_{\mathbf{x}\in\{0,1\}^n} \left[\frac{|\mathbf{x}, 0\rangle - |\mathbf{x}, 1\rangle}{\sqrt{2}}\right] \end{aligned} \tag{4.29}$$

Aplicamos U_f recordando que es un aplicación lineal que lleva $|\mathbf{x}, y\rangle$ a $|\mathbf{x}, y \oplus f(\mathbf{x})\rangle$

$$\begin{aligned} |\varphi_2\rangle = U_f \frac{1}{\sqrt{2^n}} \sum_{\mathbf{x}\in\{0,1\}^n} \left[\frac{|\mathbf{x}, 0\rangle - |\mathbf{x}, 1\rangle}{\sqrt{2}}\right] = \frac{1}{\sqrt{2^n}} \sum_{\mathbf{x}\in\{0,1\}^n} \left[\frac{U_f |\mathbf{x}, 0\rangle - U_f |\mathbf{x}, 1\rangle}{\sqrt{2}}\right] \\ = \frac{1}{\sqrt{2^n}} \sum_{\mathbf{x}\in\{0,1\}^n} \left[\frac{|\mathbf{x}, 0 \oplus f(\mathbf{x})\rangle - |\mathbf{x}, 1 \oplus f(\mathbf{x})\rangle}{\sqrt{2}}\right] = \frac{1}{\sqrt{2^n}} \sum_{\mathbf{x}\in\{0,1\}^n} |\mathbf{x}\rangle \left[\frac{|f(\mathbf{x})\rangle - |\overline{f(\mathbf{x})}\rangle}{\sqrt{2}}\right] \end{aligned} \tag{4.30}$$

donde $\overline{f(\mathbf{x})}$ es el opuesto. Recordemos ahora de expresión 4.13 que podemos poner $|f(\mathbf{x})\rangle - |\overline{f(\mathbf{x})}\rangle$ como $(-1)^{f(\mathbf{x})}(|0\rangle - |1\rangle)$, por tanto sustituyendo en la expresión anterior obtenemos

$$|\varphi_2\rangle = \left[\frac{1}{\sqrt{2^n}} \sum_{\mathbf{x}\in\{0,1\}^n} (-1)^{f(\mathbf{x})} |\mathbf{x}\rangle\right] \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right] \tag{4.31}$$

Aplicando ahora la matriz $H^{\otimes n}$ a los primeros n qubits, aprovechando la expresión 4.26 y que es lineal tenemos que

$$\begin{aligned} |\varphi_3\rangle = \left[H^{\otimes n} \frac{1}{\sqrt{2^n}} \sum_{\mathbf{x}\in\{0,1\}^n} (-1)^{f(\mathbf{x})} |\mathbf{x}\rangle\right] \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right] = \left[\frac{1}{\sqrt{2^n}} \sum_{\mathbf{x}\in\{0,1\}^n} (-1)^{f(\mathbf{x})} H^{\otimes n} |\mathbf{x}\rangle\right] \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right] \\ = \left[\frac{1}{2^n} \sum_{\mathbf{x}\in\{0,1\}^n} (-1)^{f(\mathbf{x})} \sum_{\mathbf{z}\in\{0,1\}^n} (-1)^{\langle \mathbf{z},\mathbf{x}\rangle} |\mathbf{z}\rangle\right] \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right] \end{aligned} \tag{4.32}$$

Ahora juntamos las dos potencias de (-1) teniendo en cuenta que estamos interesados por la paridad del exponente, usamos $\oplus$ la suma módulo 2.

$$|\varphi_3\rangle = \left[\frac{\sum_{\mathbf{x}\in\{0,1\}^n} \sum_{\mathbf{z}\in\{0,1\}^n} (-1)^{f(\mathbf{x}) \oplus \langle \mathbf{z},\mathbf{x}\rangle} |\mathbf{z}\rangle}{2^n}\right] \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right] \tag{4.33}$$

Finalmente nos queda medir los n primeros qubits. Nos preguntamos cuál es la probabilidad de que esté en el estado $|\mathbf{0}\rangle$, fijando $\mathbf{z} = \mathbf{0}$. Por tanto, como $\langle \mathbf{z}, \mathbf{x} \rangle = \langle \mathbf{0}, \mathbf{x} \rangle = 0$ para todo $\mathbf{x}$, tenemos que la probabilidad es

$$\left| \frac{\sum_{\mathbf{x} \in \{0,1\}^n} (-1)^{f(\mathbf{x})}}{2^n} \right|^2 \tag{4.34}$$

Observamos que depende únicamente de $f(\mathbf{x})$ por lo que podemos discutir su valor según $f(\mathbf{x})$.
Si $f(\mathbf{x})$ es constantemente 0 ó 1 tenemos que la probabilidad de estar en el estado $|\mathbf{0}\rangle$ es

$$\left| \frac{\sum_{\mathbf{x} \in \{0,1\}^n} (-1)^{f(\mathbf{x})}}{2^n} \right|^2 = \left| \frac{\sum_{\mathbf{x} \in \{0,1\}^n} (\pm 1)}{2^n} \right|^2 = \left| \frac{\pm 2^n}{2^n} \right|^2 = 1 \tag{4.35}$$

Si $f(\mathbf{x})$ es balanceada, la mitad de los términos $(-1)^{f(\mathbf{x})}$ anulan a la otra mitad y se obtiene que la probabilidad es

$$\left| \frac{\sum_{\mathbf{x} \in \{0,1\}^n} (-1)^{f(\mathbf{x})}}{2^n} \right|^2 = \left| \frac{\sum_{\mathbf{x} \in \{0,1\}^n} 0}{2^n} \right|^2 = \left| \frac{0}{2^n} \right|^2 = 0 \tag{4.36}$$

Podemos concluir que tras la medición, si encontramos el qubit en el estado $|\mathbf{0}\rangle$ la función será constante, mientras que si nos encontramos cualquier otro estado distinto de $|\mathbf{0}\rangle$ la función será balanceada. Esto se debe a que la única manera de que el estado sea $|\mathbf{0}\rangle$ es que la función sea constante.

Observamos entonces que el algoritmo cuántico necesita solamente una evaluación de la función, respecto a las $2^{n-1} + 1$ del método clásico.

4.4. Algoritmo de Búsqueda de Grover

Supongamos que tenemos un conjunto de elementos y que queremos buscar aquellos que cumplen cierta propiedad. Un método muy primitivo y poco eficiente sería ir elemento a elemento y comprobar si cumplen esta propiedad. Si consideramos un conjunto de N elementos esto nos llevaría en el peor de los casos N operaciones a realizar, comprobando la propiedad. Estos algoritmos clásicos tiene un orden de complejidad de $O(N)$ pero podemos mejorar esto con el algoritmo de búsqueda de Grover que tiene un orden de complejidad de $O(\sqrt{N})$.

Vamos a abordar el problema con funciones. Identificamos los elementos del conjunto con $\mathbf{x} \in \{0,1\}^n$ una cadena binaria. Suponemos por comodidad que $N = 2^n$ y para generalizar el problema suponemos también que hay una cantidad exacta de M elementos que cumple la propiedad que llamaremos soluciones, con $1 \leq M \leq N$. A continuación suponemos que tenemos una función $f : \{0,1\}^n \longrightarrow \{0,1\}$ que cumple que

$$f(\mathbf{x}) = \begin{cases} 1, \text{ si } \mathbf{x} \text{ es solución,} \\ 0, \text{ si } \mathbf{x} \text{ no es solución.} \end{cases} \tag{4.37}$$

Entonces, nuestro objetivo es buscar aquellas cadenas $\mathbf{x}$ que cumplen que $f(\mathbf{x}) = 1$. Para ello nos apoyaremos en n qubits que pondremos en estado de superposición de todos los estados posibles

$$|\psi\rangle = \frac{1}{\sqrt{N}} \sum_{\mathbf{x} \in \{0,1\}^n} |\mathbf{x}\rangle \tag{4.38}$$

con el propósito de, a través de una serie de operaciones, aumentar la probabilidad de encontrar los qubits en un estado asociado a una solución. Es decir, si consideramos el espacio vectorial con base $\{|\mathbf{x}\rangle \mid \mathbf{x} \in \{0,1\}^n\}$ y $|\psi\rangle$ como vector, se quiere entonces aumentar el valor de aquellas coordenadas del vector correspondientes al elemento de la base que lleve asociado una solución.

Veamos antes una interpretación geométrica de estas operaciones. Consideramos primero el vector $|\psi\rangle$ como suma de los vectores de la base y los dividimos en dos partes. Aquellas que son solución y aquellas que no.

$$|\alpha\rangle = \frac{1}{\sqrt{N-M}} \sum_{f(\mathbf{x})=0} |\mathbf{x}\rangle \qquad |\beta\rangle = \frac{1}{\sqrt{M}} \sum_{f(\mathbf{x})=1} |\mathbf{x}\rangle \tag{4.39}$$

Consideramos los vectores que los tienen por suma y lo normalizamos. Podemos ver claramente que se puede expresar $|\psi\rangle$ en función de $|\alpha\rangle$ y $|\beta\rangle$.

$$|\psi\rangle = \sqrt{\frac{N-M}{N}} |\alpha\rangle + \sqrt{\frac{M}{N}} |\beta\rangle \tag{4.40}$$

Entonces podemos poner el vector $|\psi\rangle$ en el espacio bidimensional generado por $|\alpha\rangle$ y $|\beta\rangle$ como se muestra en la figura 3.

Por tanto el objetivo se interpreta como acercar el vector $|\psi\rangle$ al eje vertical, es decir al vector $|\beta\rangle$ que son la suma de todos los estados que son solución.

Para ello se realizarán dos operaciones. Primero se realizará una simetría respecto del eje horizontal, es decir respecto de $|\alpha\rangle$. Seguido de otra simetría pero respecto del vector $|\psi\rangle$. El resultado de estas dos simetrías es por tanto el de una rotación en el plano que llamaremos por ahora G. Si suponemos que $\sqrt{\frac{N-M}{N}} = \cos\frac{\theta}{2}$ y describimos el vector $|\psi\rangle$ como $\cos\frac{\theta}{2}|\alpha\rangle + \sin\frac{\theta}{2}|\beta\rangle$, entonces se puede expresar la operación final como una rotación de ángulo θ, con $0 < \theta \leq \frac{\pi}{2}$.

Por tanto como muestra la figura 3, G lleva el vector $|\psi\rangle$ a

$$G|\psi\rangle = \cos\frac{3\theta}{2}|\alpha\rangle + \sin\frac{3\theta}{2}|\beta\rangle \tag{4.41}$$

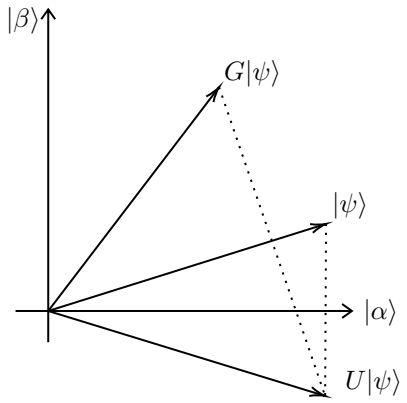


Figura 3: Representación en un plano de la iteración de Grover como dos simetrías.

Si queremos acercarnos al vector $|\beta\rangle$ bastará entonces aplicar repetidas veces G.

$$G^k |\psi\rangle = \cos \frac{(2k+1)\theta}{2} |\alpha\rangle + \sin \frac{(2k+1)\theta}{2} |\beta\rangle \tag{4.42}$$

En cuanto al número de veces que aplicaremos G, es muy fácil de determinar teniendo en cuenta que $|\psi\rangle = \sqrt{\frac{N-M}{N}} |\alpha\rangle \sqrt{\frac{M}{N}} |\beta\rangle$, por lo que para llevar $|\psi\rangle$ a $|\beta\rangle$ habría que rotar con un ángulo de $\arccos \sqrt{\frac{M}{N}}$. Entonces, sea $CI(x)$ el entero más cercano a x, las veces que tenemos que aplicar la operación G será

$$R = CI \left(\frac{\arccos \sqrt{M/N}}{\theta} \right) \tag{4.43}$$

Esta expresión nos da el valor exacto de las veces que hay que aplicar G, sin embargo podemos conseguir una expresión más simple y cómoda de trabajar con unas hipótesis adicionales. Supongamos que $M \leq \frac{N}{2}$, queremos buscar una cota superior para R, entonces tenemos que

$$\frac{\theta}{2} \geq \sin \frac{\theta}{2} = \sqrt{\frac{M}{N}} \tag{4.44}$$

entonces,

$$R \leq \left\lceil \frac{\pi}{2\theta} \right\rceil \leq \left\lceil \frac{\pi}{4} \sqrt{\frac{N}{M}} \right\rceil \tag{4.45}$$

Esto nos muestra que el algoritmo tiene una complejidad de $R = O(\sqrt{N/M})$, que requiere ese número iteraciones de G para encontrar con una alta probabilidad la solución, que es una mejora cuadrática respecto al método clásico con una complejidad de $O(N/M)$.

Veamos ahora cómo se traduce estas operaciones en términos de puertas cuánticas. Consideramos entonces el siguiente circuito:

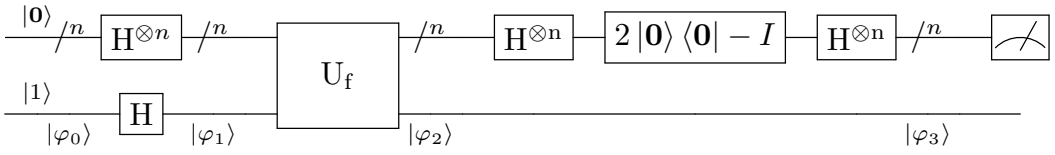


Figura 4: Circuito del algoritmo de Grover con una iteración.

Es decir, en término de matrices, sería

$$\left((H^{\otimes n} \cdot (2|\mathbf{0}\rangle\langle\mathbf{0}| - I) \cdot H^{\otimes n}) \otimes I \right) U_f (H^{\otimes n} \otimes H) |\varphi_0\rangle \tag{4.46}$$

observamos que se puede simplificar la siguiente expresión obteniendo

$$H^{\otimes n} \cdot (2 |\mathbf{0}\rangle \langle \mathbf{0}| - I) \cdot H^{\otimes n} = 2 |\psi\rangle \langle \psi| - I \tag{4.47}$$

por lo que podemos llamar como iteración de Grover a la operación $G = ((2 |\psi\rangle \langle \psi| - I) \otimes I) U_f$, que es la rotación de ángulo θ que hemos visto antes.
Empezamos como siempre con n qubits y el qubit de control en el estado $|\varphi_0\rangle = |\mathbf{0}, 1\rangle$ y lo ponemos en estado de superposición con la puerta de Hadamard $H^{\otimes n} \otimes H$ obteniendo el estado

$$|\varphi_1\rangle = \frac{1}{\sqrt{N}} \sum_{\mathbf{x} \in \{0,1\}^n} |\mathbf{x}\rangle \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right] \tag{4.48}$$

A continuación le aplicamos el oráculo U_f a $|\varphi_1\rangle$ y se obtiene

$$|\varphi_2\rangle = \frac{1}{\sqrt{N}} \sum_{\mathbf{x} \in \{0,1\}^n} (-1)^{f(\mathbf{x})} |\mathbf{x}\rangle \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right] \tag{4.49}$$

que es la misma expresión 4.31 que se llega en el algoritmo anterior. Observamos que esto es cambiar el signo de aquellos vectores $|\mathbf{x}\rangle \in \{0,1\}^n$ que cumplen $f(\mathbf{x}) = 1$, es decir está haciendo la simetría respecto del eje horizontal correspondiente al vector de no soluciones $|\alpha\rangle$.

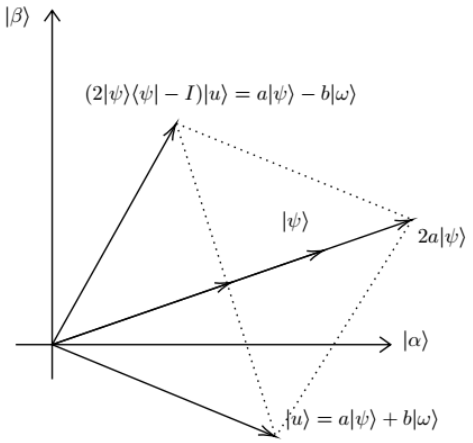
Finalmente realizaremos la simetría respecto del vector $|\psi\rangle$ que corresponde con la matriz unitaria $2 |\psi\rangle \langle \psi| - I$. Esto se puede ver fácilmente si tomamos por ejemplo un vector arbitrario $|u\rangle$. Consideramos primero $|\omega\rangle$ que cumple que $\{|\psi\rangle, |\omega\rangle\}$ forma una base ortonormal. Entonces podemos expresar $|u\rangle = a |\psi\rangle + b |\omega\rangle$, por tanto tenemos que $(2 |\psi\rangle \langle \psi| - I) |u\rangle$ es

$$\begin{aligned} (2 |\psi\rangle \langle \psi| - I) (a |\psi\rangle + b |\omega\rangle) &= 2 |\psi\rangle \langle \psi| (a |\psi\rangle + b |\omega\rangle) - (a |\psi\rangle + b |\omega\rangle) \\ &= 2a |\psi\rangle \langle \psi| (|\psi\rangle) + 2b |\psi\rangle \langle \psi| (|\omega\rangle) - (a |\psi\rangle + b |\omega\rangle) \\ &= 2a \langle \psi \mid \psi\rangle |\psi\rangle + 2b \langle \psi \mid \omega\rangle |\psi\rangle - (a |\psi\rangle + b |\omega\rangle) \end{aligned} \tag{4.50}$$

Como $\{|\psi\rangle, |\omega\rangle\}$ es una base ortonormal, se cumple que $\langle \psi \mid \psi\rangle = 1$ y que $\langle \psi \mid \omega\rangle = 0$, por tanto

$$\begin{aligned} (2 |\psi\rangle \langle \psi| - I) (a |\psi\rangle + b |\omega\rangle) &= 2a |\psi\rangle - (a |\psi\rangle + b |\omega\rangle) \\ &= a |\psi\rangle - b |\omega\rangle \end{aligned} \tag{4.51}$$

Por tanto se ve que es una simetría respecto del vector $|\psi\rangle$. Además podemos ver que $|\psi\rangle \langle \psi|$ es una proyección sobre $|\psi\rangle$.



4.5. Algoritmo de Periodicidad de Simon

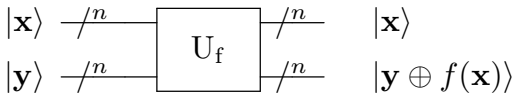
Veremos ahora el algoritmo de Simon que trata de buscar patrones que se repiten en funciones. Combinaremos en el algoritmo procedimientos cuánticos que hemos visto y también operaciones clásicas.

Consideramos entonces una función $f : \{0,1\}^n \longrightarrow \{0,1\}^n$, que como siempre podemos evaluar pero no conocemos su definición. Además sabemos que existe una cadena binaria $\mathbf{c} \in \{0,1\}^n$ tal que para toda cadena $\mathbf{x}, \mathbf{y} \in \{0,1\}^n$ tenemos que se cumple

$$f(\mathbf{x}) = f(\mathbf{y}) \text{ si y sólo si } \mathbf{x} = \mathbf{y} \oplus \mathbf{c} \tag{4.52}$$

donde $\oplus$ es la suma módulo 2 aplicado dígito a dígito. Es decir, que se cumple que $f(\mathbf{y}) = f(\mathbf{y} \oplus \mathbf{c})$. Esto es que los valores de f se repiten en un determinado patrón y llamaremos entonces $\mathbf{c}$ como el periodo de la función, que es el objetivo del algoritmo.

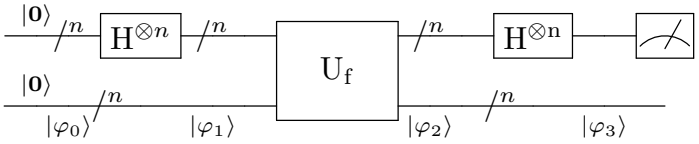
Sea entonces U_f la matriz unitaria que define el oráculo, que lleva $|\mathbf{x}, \mathbf{y}\rangle$ a $|\mathbf{x}, \mathbf{y} \oplus f(\mathbf{x})\rangle$, que podemos ver de nuevo que es su propia inversa.



Observamos que en este caso tenemos $2n$ qubits y si ponemos los últimos n qubits en el estado $|\mathbf{y}\rangle = |\mathbf{0}\rangle$, nos permitiría evaluar la función con el oráculo

$$U_f |\mathbf{x}, \mathbf{0}\rangle = |\mathbf{x}, \mathbf{0} \oplus f(\mathbf{x})\rangle = |\mathbf{x}, f(\mathbf{x})\rangle \tag{4.53}$$

Teniendo en cuenta esto, consideramos el siguiente circuito y veamos las operaciones paso a paso.



Esto en término de matrices sería

$$(H^{\otimes n} \otimes I) U_f (H^{\otimes n} \otimes I) |\mathbf{0}, \mathbf{0}\rangle \tag{4.54}$$

Comenzamos con los qubits en el estado $|\varphi_0\rangle = |\mathbf{0}, \mathbf{0}\rangle$ y lo ponemos en estado de superposición con la matriz de Hadamard. Recordamos que el resultado de la expresión 4.27, sabemos que esto es

$$|\varphi_1\rangle = \frac{1}{\sqrt{2^n}} \sum_{\mathbf{x} \in \{0,1\}^n} |\mathbf{x}, \mathbf{0}\rangle \tag{4.55}$$

Aplicamos el oráculo U_f y recordando que es lineal tenemos,

$$|\varphi_2\rangle = \frac{1}{\sqrt{2^n}} \sum_{\mathbf{x} \in \{0,1\}^n} |\mathbf{x}, f(\mathbf{x})\rangle \tag{4.56}$$

Finalmente, aplicamos de nuevo la matriz de Hadamard teniendo en cuenta la expresión 4.26 y obtenemos

$$|\varphi_3\rangle = \frac{\sum_{\mathbf{x} \in \{0,1\}^n} \sum_{\mathbf{z} \in \{0,1\}^n} (-1)^{\langle \mathbf{z}, \mathbf{x} \rangle} |\mathbf{x}, f(\mathbf{x})\rangle}{2^n} \tag{4.57}$$

Observamos entonces que en el sumatorio, para cada $\mathbf{x}$ y para cada $\mathbf{z}$ tenemos que $|\mathbf{z}, f(\mathbf{x})\rangle$ coincide con $|\mathbf{z}, f(\mathbf{x} \oplus \mathbf{c})\rangle$ por la periodicidad de la función. Por tanto si agrupamos los pares, sus coeficientes serán

$$\begin{aligned} \frac{(-1)^{\langle \mathbf{z},\mathbf{x} \rangle} + (-1)^{\langle \mathbf{z},\mathbf{x} \oplus \mathbf{c} \rangle}}{2} &= \frac{(-1)^{\langle \mathbf{z},\mathbf{x} \rangle} + (-1)^{\langle \mathbf{z},\mathbf{x} \rangle \oplus \langle \mathbf{z},\mathbf{c} \rangle}}{2} \\ = \frac{(-1)^{\langle \mathbf{z},\mathbf{x} \rangle} + (-1)^{\langle \mathbf{z},\mathbf{x} \rangle}(-1)^{\langle \mathbf{z},\mathbf{c} \rangle}}{2} &= \frac{(-1)^{\langle \mathbf{z},\mathbf{x} \rangle}(1 + (-1)^{\langle \mathbf{z},\mathbf{c} \rangle})}{2} \end{aligned} \tag{4.58}$$

Entonces, si discutimos la expresión según el valor de $\langle \mathbf{z}, \mathbf{c} \rangle$ tenemos que si $\langle \mathbf{z}, \mathbf{c} \rangle = 1$, la expresión anterior se anula. Si $\langle \mathbf{z}, \mathbf{c} \rangle = 0$ obtenemos el valor ± 1. Por tanto si medimos los n primeros qubits tras el circuito, obtendremos las cadena binarias $\mathbf{z}$ que cumplen que $\langle \mathbf{z}, \mathbf{c} \rangle = 0$. Lo que nos queda por hacer es sólamente plantear una ecuación lineal con las cadenas resultantes para obtener $\mathbf{c}$.

4.6. Algoritmo de Factorización de Shor

El problema de factorizar un número compuesto tiene su interés en la criptografía, ya que este se basa en que no hay un algoritmo eficiente que resuelva el problema. Sin embargo la computación cuántica nos ofrece un algoritmo que permite descomponer un número compuesto en sus factores primos de manera exponencialmente más rápido que el mejor algoritmo clásico que se conoce hasta la fecha.

Antes de entrar en detalle con el algoritmo veremos una herramienta que es esencial para el proceso de este algoritmo, además de otros también. Se trata de la Transformada Cuántica de Fourier, que nos permite aproximar el valor de la fase de un estado de un qubit, algo realmente importante que nos permite resolver por ejemplo el problema de encontrar el orden de un elemento y también al problema de factorizar un número compuesto.

Veamos primero en detalle la transformada cuántica de Fourier. Recordemos primero la transformada discreta de Fourier, que lleva un vector de complejos $(x_0, \ldots, x_{N-1})$ a otro de misma dimensión $y_0, \ldots, y_{N-1}$ definido cada componente como

$$y_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j e^{2\pi i j k/N} \tag{4.59}$$

Por tanto queremos que la transformada cuántica de Fourier actúe sobre una base ortonormal $\{|j\rangle \mid 0 \leq j \leq N-1\}$ de manera que a cada elemento de la base

$$|j\rangle \longrightarrow \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi i j k/N} |k\rangle \tag{4.60}$$

Entonces, si se toma un vector arbitrario, la transformada cuántica de Fourier lo llevará a

$$\sum_{j=0}^{N-1} x_j |j\rangle \longrightarrow \sum_{k=0}^{N-1} y_k |k\rangle \tag{4.61}$$

donde las amplitudes y_k son la transformada discreta de Fourier aplicado a x_j. Esto es la definición general que podemos dar sobre la transformada cuántica de Fourier, sin embargo hay una expresión con la que nos resultará más cómoda trabajar posteriormente. Además nos permitirá demostrar fácilmente que la transformada cuántica de Fourier no es sólo lineal sino también unitaria, que es importante que cumpla esto último.

Tomamos a partir de ahora $N = 2^n$ para n entero, y la base ortonormal $\{|0\rangle, \ldots, |2^n - 1\rangle\}$ representado el estado de n qubits. Entonces para cada $|j\rangle$ podemos escribirlo en su representación binaria con $j \equiv j_1 j_2 \ldots j_n$ de manera que $j = j_1 2^{n-1} + j_2 2^{n-2} + \ldots + j_n 2^0$. Consideraremos también la notación binaria para fracciones $0.j_1 j_2 \ldots j_m$ para representar el valor $j_1 2^{-1} + j_2 2^{-2} + \ldots + j_m^{-m}$. Tomamos entonces un elemento de la base $|j\rangle$ y le aplicamos la transformada cuántica de Fourier

$$\begin{aligned} |j\rangle &\to \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} e^{2\pi i j k/2^n} |k\rangle \\ &= \frac{1}{\sqrt{2^n}} \sum_{k_1=0}^{1} \cdots \sum_{k_n=0}^{1} e^{2\pi i j \left(\sum_{l=1}^{n} k_l 2^{-l}\right)} |k_1 \ldots k_n\rangle \end{aligned} \tag{4.62}$$

donde hemos sustituido k por su expresión en binario $k_1 \ldots k_n$ y $\frac{k}{2^n} = \frac{k_1 2^{n-1} + k_2 2^{n-2} k_n 2^0}{2^n} = \sum_{l=1}^{n} k_l 2^{-l}$.

Separamos ahora cada vector por el producto tensorial y simplicamos la expresión

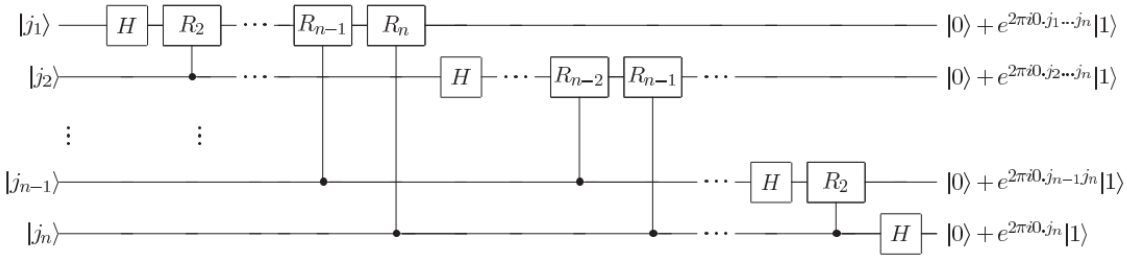
$$\begin{aligned}
|j\rangle &\to \frac{1}{\sqrt{2^n}} \sum_{k_1=0}^{1} \cdots \sum_{k_n=0}^{1} \left(\prod_{l=1}^{n} e^{2\pi i j k_l 2^{-l}} \right) |k_1 \dots k_n\rangle \\
&= \frac{1}{\sqrt{2^n}} \sum_{k_1=0}^{1} \cdots \sum_{k_n=0}^{1} \bigotimes_{l=1}^{n} e^{2\pi i j k_l 2^{-l}} |k_l\rangle \\
&= \frac{1}{\sqrt{2^n}} \bigotimes_{l=1}^{n} \left[\sum_{k_l=0}^{1} e^{2\pi i j k_l 2^{-l}} |k_l\rangle \right] \\
&= \frac{1}{\sqrt{2^n}} \bigotimes_{l=1}^{n} \left[|0\rangle + e^{2\pi i j 2^{-l}} |1\rangle \right]
\end{aligned} \tag{4.63}$$

Vamos a analizar por un momento en detalle el valor de $e^{2\pi i j 2^{-l}}$. Si tomamos por ejemplo el valor de $l = 1$ y expresamos j en binario tenemos que $\frac{j}{2^1} = \frac{j_1 2^{n-1} + j_2 2^{n-2} + \ldots + j_n 2^0}{2} = j_1 2^{n-2} + j_2 2^{n-3} + \ldots + j_{n-1} 2^0 + j_n 2^{-1}$ lo que podemos separar $e^{2\pi i j 2^{-l}} = e^{2\pi i \left(j_1 2^{n-2} + j_2 2^{n-3} + \ldots + j_{n-1} 2^0\right)} \cdot e^{2\pi i j_n 2^{-1}}$. Ahora teniendo en cuenta que $\left(j_1 2^{n-2} + j_2 2^{n-3} + \ldots + j_{n-1} 2^0\right)$ es un número entero, tenemos que $e^{2\pi i j 2^{-l}} = e^{2\pi i} \cdot e^{2\pi i j_n 2^{-1}} = 1 \cdot e^{2\pi i 0.j_n}$ donde hemos expresado $j_n 2^{-1}$ con su expresión binaria para fracciones. Entonces en general para un valor en concreto de $l = m$ podemos expresar $e^{2\pi i j 2^{-m}} = e^{2\pi i 0.j_{n-m+1} \cdots j_n}$.

$$\begin{aligned}
|j\rangle &\to \frac{1}{\sqrt{2^n}} \bigotimes_{l=1}^{n} \left[|0\rangle + e^{2\pi i j 2^{-l}} |1\rangle \right] \\
&= \frac{\left(|0\rangle + e^{2\pi i 0.j_n} |1\rangle\right) \left(|0\rangle + e^{2\pi i 0.j_{n-1} j_n} |1\rangle\right) \cdots \left(|0\rangle + e^{2\pi i 0.j_1 j_2 \ldots j_n} |1\rangle\right)}{\sqrt{2^n}}
\end{aligned} \tag{4.64}$$

Esta última expresión nos resultará de mucha utilidad a pesar de que a primera vista parece incómodo para trabajar. Esto nos permitirá construir un circuito de manera muy intuitiva para aplicar la transformada cuántica de Fourier. Para ello consideramos primero la puerta R_k cuya matriz es

$$R_k = \begin{bmatrix} 1 & 0 \\ 0 & e^{2\pi i / 2^k} \end{bmatrix} \tag{4.65}$$



Comenzaremos con el estado $|j_1 j_2 \ldots j_n\rangle$. Si le aplicamos la puerta de Hadamard al primero estado, podemos expresar este como

$$\frac{1}{\sqrt{2}} \left(|0\rangle + e^{2\pi i 0.j_1} |1\rangle\right) |j_2 \ldots j_n\rangle \tag{4.66}$$

ya que el valor de $e^{2\pi i 0.j_1}$ es

$$\begin{cases} j_1 = 0 \longrightarrow e^{2\pi i \cdot 0} = e^0 = 1 \\ j_1 = 1 \longrightarrow e^{2\pi i \cdot 0.1} = e^{2\pi i \cdot 2^{-1}} = e^{\pi i} = -1 \end{cases} \tag{4.67}$$

Le aplicamos a continuación la puerta controlada R_2 al qubit actual como qubit objetivo, y el qubit en el estado $|j_2\rangle$ como control. Esto es

$$\begin{cases} j_2 = 0 \longrightarrow \frac{1}{\sqrt{2}} \left(|0\rangle + e^{2\pi i 0.j_1} |1\rangle\right) = \frac{1}{\sqrt{2}} \left(|0\rangle + e^{2\pi i 0.j_1 0} |1\rangle\right) \\ j_2 = 1 \longrightarrow \frac{1}{\sqrt{2}} \left(|0\rangle + e^{2\pi i 0.j_1} \cdot e^{2\pi i/2^2} |1\rangle\right) = \frac{1}{\sqrt{2}} \left(|0\rangle + e^{2\pi i 0.j_1 1} \cdot |1\rangle\right) \end{cases} \tag{4.68}$$

Por tanto, tenemos el estado

$$\frac{1}{\sqrt{2}} \left(|0\rangle + e^{2\pi i 0.j_1 j_2} |1\rangle\right) |j_2 \dots j_n\rangle \tag{4.69}$$

Continuamos aplicando las puertas controladas R_3 hasta R_n que añade un nuevo bit a la fase del estado, acabando finalmente en

$$\frac{1}{\sqrt{2}} \left(|0\rangle + e^{2\pi i 0.j_1 j_2 \dots j_n} |1\rangle\right) |j_2 \dots j_n\rangle \tag{4.70}$$

De manera análoga realizamos el mismo proceso con el resto de qubits lo que nos permite obtener el estado final

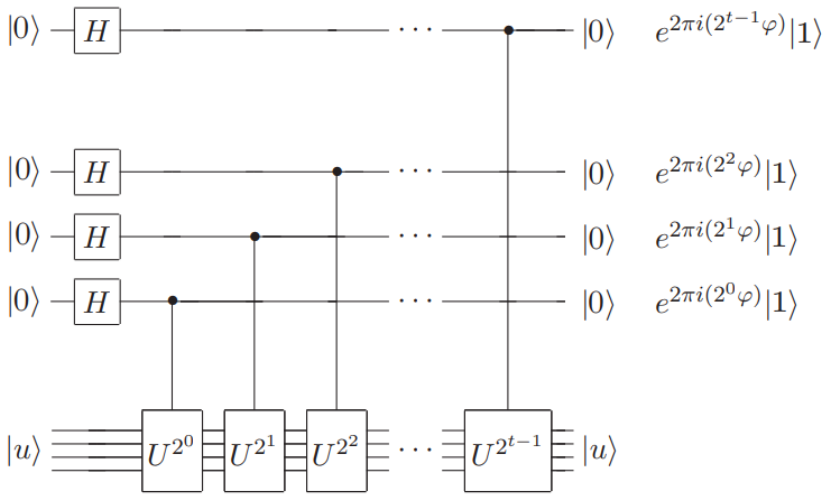
$$\frac{1}{\sqrt{2^n}} \left(|0\rangle + e^{2\pi i 0.j_1 j_2 \dots j_n} |1\rangle\right) \left(|0\rangle + e^{2\pi i 0.j_2 \dots j_n} |1\rangle\right) \dots \left(|0\rangle + e^{2\pi i 0.j_n} |1\rangle\right) \tag{4.71}$$

Finalmente realizamos una operación de intercambio con la puerta de intercambio para obtener el orden adecuado y de esta manera obtenemos la transformada cuántica de Fourier

$$\frac{1}{\sqrt{2^n}} \left(|0\rangle + e^{2\pi i 0.j_n} |1\rangle\right) \left(|0\rangle + e^{2\pi i 0.j_{n-1} j_n} |1\rangle\right) \dots \left(|0\rangle + e^{2\pi i 0.j_1 j_2 \dots j_n} |1\rangle\right) \tag{4.72}$$

Con esto terminamos en el estado que buscábamos además de demostrar que la transformada cuántica de Fourier es unitaria pues está compuesto de operaciones unitarias.

Ahora veremos que la transformación cuántica de Fourier es esencial para el proceso de la estimación de la fase. Este último consiste en considerar una aplicación unitaria U que tiene un autovector $|u\rangle$ con autovalor $e^{2\pi\varphi}$ donde se desconoce el valor de φ. Tenemos el objetivo de estimar el valor de φ y para ello dispondremos de el oráculo que nos permitirá realizar la operación controlada de U^{2^j}.
Comenzaremos con dos registros de qubits, o dos conjuntos de qubits. El primero registro tendrá t qubits en el estado $|\mathbf{0}\rangle$ con t el número de bits que contendrá la aproximación de φ.
En el segundo registro se hallarán los qubits en el estado $|u\rangle$ con el número de qubits necesarios para almacenar $|u\rangle$. Entonces, la estimación de fase consistirá en dos partes. Primero le aplicaremos el circuito siguiente.



Primero aplicaremos la puerta de Hadamard que llevará a cada qubit al estado

$$\frac{|0\rangle + |1\rangle}{\sqrt{2}} \tag{4.73}$$

Se le aplicará posteriormente las puertas controladas U^{2^j} con $0 \leq j \leq 2^{t-1}$. Si tomamos los qubits afectados por la primera puerta controlada $U^{2^0} = U$ tendremos

$$\begin{aligned} {}^{C}U\left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}|u\rangle\right) &= {}^{C}U\left(\frac{|0,u\rangle + |1,u\rangle}{\sqrt{2}}\right) = \frac{1}{\sqrt{2}}\left({}^{C}U|0,u\rangle + {}^{C}U|1,u\rangle\right) \\ &= \frac{1}{\sqrt{2}}\left(|0,u\rangle + e^{2\pi i\varphi}|1,u\rangle\right) = \frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i(2^0\varphi)}|1\rangle\right)|u\rangle \end{aligned} \tag{4.74}$$

Por tanto, deducimos de las demás puertas, el estado final de los qubits del primer registro serán

$$\frac{1}{\sqrt{2^t}}\left(|0\rangle + e^{2\pi i 2^{t-1}\varphi}|1\rangle\right)\left(|0\rangle + e^{2\pi i 2^{t-2}\varphi}|1\rangle\right)\ldots\left(|0\rangle + e^{2\pi i 2^0\varphi}|1\rangle\right) \tag{4.75}$$

Por último aplicaremos la inversa de la transformada cuántica de Fourier, que consiste en darle la vuelta al circuito de la Figura 4.6. Supongamos que la mejor aproximación de φ en t bits es $\varphi \simeq 0.\varphi_1\ldots\varphi_t$, entonces reescribiendo la expresión anterior del estado en binario tendremos

$$\frac{1}{\sqrt{2^t}}\left(|0\rangle + e^{2\pi i 0.\varphi_t}|1\rangle\right)\left(|0\rangle + e^{2\pi i 0.\varphi_{t-1}\varphi_t}|1\rangle\right)\ldots\left(|0\rangle + e^{2\pi i 0.\varphi_1\varphi_2\ldots\varphi_t}|1\rangle\right) \tag{4.76}$$

Esta expresión es exactamente la transformada cuántica de Fourier de $|\varphi_1\ldots\varphi_t\rangle$, por tanto aplicando la inversa obtendremos los valores de $\varphi \simeq 0.\varphi_1\ldots\varphi_t$. Para ello mediremos los t qubits del primer registro obteniendo una buena aproximación del valor de φ.

Esta herramienta nos resolverá el problema de encontrar el orden de un elemento y también al problema de factorizar un número compuesto, que son equivalentes entre sí. Veremos primero el problema de encontrar el orden de un elemento y veremos posteriormente que el problema de factorización lo podremos transformar en un problema de encontrar el orden.

Consideremos entonces dos enteros positivos x y N tal que $x < N$ sin factores comunes, llamaremos entonces al *orden* de x *mod* N al menor entero positivo tal que $x^r = 1$ *mod* N. Entonces, el algoritmo cuántico para encontrar el orden es simplemente aplicar el algoritmo de estimación de fase con el operador unitario

$$U|y\rangle = |xy \ mod \ N\rangle \tag{4.77}$$

con $y \in \{0,1\}^L$ siendo L el número de qubits necesarios para representar N en binario. Consideramos entonces el estado

$$|u_s\rangle = \frac{1}{\sqrt{r}}\sum_{k=0}^{r-1} e^{-2\pi isk/r}|x^k \ mod \ N\rangle \tag{4.78}$$

con $0 \leq s \leq r-1$, que son autovectores de U, ya que

$$\begin{aligned} U|u_s\rangle &= U\left(\frac{1}{\sqrt{r}}\sum_{k=0}^{r-1} e^{-2\pi isk/r}|x^k \ mod \ N\rangle\right) = \frac{1}{\sqrt{r}}\sum_{k=0}^{r-1} e^{-2\pi isk/r}U|x^k \ mod \ N\rangle \\ &= \frac{1}{\sqrt{r}}\sum_{k=0}^{r-1} e^{-2\pi isk/r}|x^{k+1} \ mod \ N\rangle \end{aligned} \tag{4.79}$$

Desarrollamos la expresión y sacando factor común obtenemos

$$\begin{aligned} U|u_s\rangle &= \frac{1}{\sqrt{r}}\left(e^{\frac{-2\pi is}{r}0}|x^1 \ mod \ N\rangle + e^{\frac{-2\pi is}{r}1}|x^2 \ mod \ N\rangle + \ldots + e^{\frac{-2\pi is}{r}(r-1)}|x^r \ mod \ N\rangle\right) \\ &= e^{2\pi is/r}\left[\frac{1}{\sqrt{r}}\left(e^{\frac{-2\pi is}{r}1}|x^1 \ mod \ N\rangle + e^{\frac{-2\pi is}{r}2}|x^2 \ mod \ N\rangle + \ldots + e^{\frac{-2\pi is}{r}r}|x^r \ mod \ N\rangle\right)\right] \end{aligned} \tag{4.80}$$

Teniendo en cuenta ahora que $e^{-2\pi is} = 1 = e^0$ y que $x^r = 1 \; mod \; N = x^0 \; mod \; N$, tenemos que

$$\begin{aligned} U\left|u_s\right\rangle &= e^{2\pi is/r}\left[\frac{1}{\sqrt{r}}\left(e^{\frac{-2\pi is}{r}1}\left|x^1 \; mod \; N\right\rangle + e^{\frac{-2\pi is}{r}2}\left|x^2 \; mod \; N\right\rangle + \ldots + e^{\frac{-2\pi is}{r}0}\left|x^0 \; mod \; N\right\rangle\right)\right] \\ &= e^{2\pi is/r}\left[\frac{1}{\sqrt{r}}\sum_{k=0}^{r-1} e^{-2\pi isk/r}\left|x^k \; mod \; N\right\rangle\right] = e^{2\pi is/r}\left|u_s\right\rangle \end{aligned} \tag{4.81}$$

Por tanto aplicando el algoritmo de estimación fase, obtendremos una aproximación del valor de $\varphi \simeq s/r$, del que podemos obtener fácilmente el valor de r.

Una vez que sabemos hallar el orden de un elemento, usaremos esta herramienta y dos teoremas para resolver el problema de factorización. La idea es reducir el problema de factorización en el problema de encontrar el orden de un elemento. Vamos a desarrollar primero algunos conceptos necesarios de teoría de números.

Presentamos primero el algoritmo de Euclides para encontrar el máximo común divisor de dos números. Supongamos entonces a, b dos enteros de manera que $a > b$.

Teorema 4.1. *Sea entonces r el resto de la división de a entre b. Si se da que $r = 0$, tenemos que se cumple*

$$gcd(a, b) = gcd(b, r) \tag{4.82}$$

donde $gcd(a, b)$ denota el máximo común divisor de a y b.

El algoritmo de Euclides consiste en dividir a entre b resultando en k_1 como cociente y un resto r_1, $a = k_1 b + r_1$. Por el teorema 4.1 tenemos que $gcd(a, b) = gcd(b, r_1)$. Si repetimos el proceso con b dividido por r_1 tenemos de nuevo que $b = k_2 r_1 + r_2$. Y de nuevo por el teorema se tiene $gcd(a, b) = gcd(b, r_1) = gcd(r_1, r_2)$. Siguiendo este proceso repetidas veces hasta obtener un resto 0, $r_m = k_{m+1} r_{m+1}$ para algún m. Entonces tenemos que $gcd(a, b) = gcd(r_m, r_{m+1}) = r_{m+1}$.

Veamos ahora el coste de este algoritmo. Supongamos que a y b pueden ser representadas por cadenas binarias de longitud L. Como los cocientes k_i y restos r_i son menores que a, lo podremos representar con cadenas de longitud L. Veamos ahora que se cumple que para cada i, $r_{i+2} \leq r_i/2$.

- Supongamos que $r_{i+1} \leq r_i/2$, entonces como $r_{i+2} \leq r_{i+1}$ tenemos que $r_{i+2} \leq r_i/2$.
- Supongamos que $r_{i+1} > r_i/2$. Como $r_i = 1 \cdot r_{i+1} + r_{i+2}$ tenemos que $r_{i+2} = r_i - r_{i+1} \leq r_i/2$.

Como $r_{i+2} \leq r_i/2$, el resto se reduce al menos a la mitad cada dos iteraciones. Como el algoritmo para cuando el resto se anula, tenemos que como máximo el proceso de división y resto se tiene que iterar como máximo un número de $2\lceil \log a \rceil = O(L)$ veces. Como cada proceso de división y resto llevan $O(L^2)$ operaciones, tenemos que el coste total del algoritmo de Euclides es de $O(L^3)$.

Veamos ahora los dos teoremas que nos ayudarán en reducir el problema de factorización.

Teorema 4.2. *Supongamos que N es un número compuesto representado en L bits y x es la solución no trivial de $x^2 = 1 \; mod \; N$ con $1 \leq x \leq N$, es decir que $x \neq \pm 1 \; mod \; N$. Entonces se da que al menos uno de $gcd(x-1, N)$ y $gcd(x+1, N)$ es un factor primo no trivial de N que puede ser calculado en un número de $O(L^3)$ operaciones.*

Demostración. Como $x^2 = 1 \; mod \; N$, entonces N divide a $x^2 - 1 = (x+1)(x-1)$ por tanto N tiene que tener factor común con $(x+1)$ ó con $(x-1)$. Tenemos por hipótesis que $1 \leq x \leq N$ por tanto $x - 1 < x + 1 < N$, el factor común no puede ser N. Empleando entonces el algoritmo de Euclides que hemos visto antes para calcular $gcd(x-1, N)$ y $gcd(x+1, N)$, obteniendo un factor no trivial de N en $O(L^3)$ operaciones. □

Definimos la función de Euler $\varphi(n)$ como el número de enteros positivos menores que n coprimo con él. Si p es primo, todos los números menores a él son coprimos con p, entonces $\varphi(p) = p - 1$. Si se toma p^α una potencia de p, los únicos coprimos con p^α son sus múltiplos $p, 2p, 3p, \ldots, (p^{\alpha-1} - 1)p$, por tanto

$$\varphi(p^\alpha) = (p^\alpha - 1) - (p^{\alpha-1} - 1) = p^{\alpha-1}(p-1) \tag{4.83}$$

Lema 4.1. *Sea p un primo impar. Sea 2^d la mayor potencia de 2 que divide a $\varphi(p^\alpha)$, con $\varphi(n)$ siendo la función de Euler y p^α una potencia de p. Entonces con una probabilidad exacta de $1/2$, se tiene que 2^d divide el orden de un $x \bmod p^\alpha$ escogido aleatoriamente de $\mathbb{Z}^*_{p^\alpha}$.*

Demostración. Como p es primo impar, tenemos que $\varphi(p^\alpha) = p^{\alpha-1}(p-1)$ es par y por tanto $d \geq 1$. Sabemos que $\mathbb{Z}^*_{p^\alpha}$ es cíclico, es decir, existe g elemento de $\mathbb{Z}^*_{p^\alpha}$ tal que, cualquier elemento se puede escribir como $g^k \ mod \ p^\alpha$ para k de 1 a $\varphi(p^\alpha)$. Observamos que $g^{\varphi(p^\alpha)} = 1 \ mod \ p^\alpha$ que es el elemento nulo de $\mathbb{Z}^*_{p^\alpha}$. Sea entonces r el orden de $g^k \ mod \ p^\alpha$ y consideramos dos casos

- Suponemos primero que k es impar. Como $g^{kr} = 1 = g^{\varphi(p^\alpha)} \ mod \ p^\alpha$, se tiene que $\varphi(p^\alpha) \mid kr$. Y como k es impar y 2^d que divide a $\varphi(p^\alpha)$ se tiene que $2^d \mid r$.
- Suponemos ahora que k es par. Entonces $g^{k\varphi(p^\alpha)/2} = \left(g^{\varphi(p^\alpha)}\right)^{k/2} = 1^{k/2} = 1 \ mod \ p^\alpha$. Entonces $r \mid \varphi(p^\alpha)/2$ de donde podemos deducir que 2^d no divide a r.

Por tanto se tiene que si k es impar $2^d \mid r$ y si k es par se tiene $2^d \nmid r$. □

Teorema 4.3. *Consideramos $N = p_1^{\alpha^1} \dots p_m^{\alpha^m}$ la factorización prima de un número compuesto impar positivo. Sea x un entero elegido uniformente aleatorio de $\mathbb{Z}^*_N$, tal que $1 \leq x \leq N-1$ y x es coprimo con N. Sea r el orden de $x \bmod N$. Entonces la probabilidad*

$$P(r \ es \ par \ y \ x^{r/2} \neq -1 \ mod \ N) \geq 1 - \frac{1}{2^m} \tag{4.84}$$

Demostración. Vamos a probar que

$$P(r \text{ es impar ó } x^{r/2} = -1 \ mod \ N) \leq \frac{1}{2^m} \tag{4.85}$$

Sabemos que $\mathbb{Z}^*_N$ es isomorfo a producto cartesiano $\prod_j \mathbb{Z}^*_{p_j^{\alpha_j}}$. Escoger x aleatoriamente es igual a escoger las coordenadas $x_j \in \mathbb{Z}^*_{p_j^{\alpha_j}}$ de manera independiente con $x = x_j \ mod \ p_j^{\alpha_j}$.
Consideramos entonces r_j el orden de $x_j \ mod \ p_j^{\alpha_j}$, 2^{d_j} la mayor potencia de 2 que divide a r_j y 2^d la mayor potencia de 2 que divide a r.
Vamos a probar que para que r sea impar ó $x^{r/2} = -1 \ mod \ N$ tiene que ocurrir que para todo j, d_j tome el mismo valor.

- Supongamos que r es impar. Como r_j es el orden de las coordenadas de $(x_j)_j$ y r es el orden de x, tenemos que $r_j \mid r$ para cada j. Por tanto r_j es impar y la mayor potencia que lo divide es 1 y se llega a que d_j es 0.
- Supongamos ahora que r es par, entonces no queda otra que $x^{r/2} = -1 \ mod \ N$. Se tiene por tanto también que para cada j, $x^{r/2} = -1 \ mod \ p_j^{\alpha_j}$. Como $x = x_j \ mod \ p_j^{\alpha_j}$ tenemos que $(x_j)^{r/2} = -1 \ mod \ p_j^{\alpha_j}$.
  Si se cumple que $r_j \mid r/2$ entonces existe un k tal que $r/2 = k \cdot r_j$ y $(x_j)^{r/2} = (x_j)^{k \cdot r_j} = (x_j^{r_j})^k = 1^k = 1 \ mod \ p_j^{\alpha_j}$, llegando a contradicción. Entonces $r_j \nmid r/2$. Además tenemos que $r_j \mid r$.
  De aquí podemos concluir que para cada j se tiene que $r_j = 2$ ó $r_j = r$.
  Si algún j, $r_j = r$, es fácil ver que para ese j, $d_j = d$. Pero si para algún j se tiene que $r_j = 2$, entonces como ese $r_j \nmid r/2$ tenemos que $r/2$ es impar. Como $r = (r/2) \cdot 2)$ la mayor potencia de 2 que lo divide es 2 con $d = 1$. Como $r_j = 2$ la mayor potencia de 2 que le divide es 2 con $d_j = 1$, por lo que tanto se tiene que $d_j = d$.

Ahora sabemos que para que r sea impar ó $x^{r/2} = -1 \ mod \ N$ tiene que ocurrir que para todo j, d_j tome el mismo valor y por el lema 4.1 tenemos que con una probabilidad de $1/2$ pasa para un sólo d_j y entonces para que tome el mismo valor, la probabilidad es de a lo sumo $1/2^m$. □

Entonces a partir de esto teoremas podemos sacar un algoritmo para factorizar consiste en los siguientes pasos. Todos los pasos se pueden realizar en un ordenador clásico de manera eficiente menos el paso número 4 es el que realizaremos con un algoritmo cuántico.

1. Si N es par, devolvemos el factor 2.
2. Determinar si se cumple $N = a^b$ para enteros $a \geq 1$ y $b \geq 2$. Devolver el factor a en caso positivo.
3. Elegir aleatoriamente x entre 1 y $N-1$. Si $gcd(x, N) > 1$ entonces devolvemos el factor $gcd(x, N)$.
4. Aplicamos el algoritmo cuántico para encontrar el orden r de $x \; mod \; N$.
5. Si r es par y $x^{r/2} \neq -1 \; mod \; N$, comprobar si $gcd(x^{r/2} - 1, N)$ y $gcd(x^{r/2} + 1, N)$ son factores no triviales devolviendo el el factor resultante. En caso contrario el algoritmo falla.

5. Conclusión

En esta sección no mostraremos una conclusión como tal, pues por la naturaleza del trabajo, no se lleva a un resultado que exija su presentación en un conclusión.
Sin embargo aprovecharemos esta parte para mostrar algunas impresiones propias sobre el trabajo en vista a lo que se ha hecho.

Teniendo en cuenta entonces el área de estudio, podemos decir que ha sido muy interesante ver tanto, soluciones a algunas cuestiones de la computación que se han visto en la propia carrera como las técnicas aplicadas en ella. A pesar de lo sencillo que puede resultar la álgebra lineal en principio, hemos visto que es potente permitiendo a uno resolver incluso problemas con una complejidad elevada.

Sin embargo, lamentamos la falta de tanto tiempo como conocimientos para entrar en detalles más profundos y complejos, ya sea en cuestiones de la mecánica cuántica que es de vital importancia, o de cualquier cuestión propia de la computación cuántica que no hemos entrado en detalle. Damos de ejemplo la existencia de puertas universales, como en la última parte donde se desarrollan ideas de la teoría de números. A pesar de todos esto, hacemos notar que ambos libros ofrece fuentes de información para los temas más complejos, ya sea referenciados o en el propio libro.

Además también queremos mencionar sobre las implementaciones de los algoritmos. Debido tanto a la falta de conocimientos y también a la falta de recursos las implementaciones están realizadas con un número bajo de qubits por la naturaleza de las simulaciones, donde un número elevado de qubits conlleva un incremento exponencial en el uso de los recursos. Y también queremos hacer notar, quizás lo único que podríamos concluir en este trabajo, es que en las pocas ejecuciones del algoritmo en un ordenador cuántico real hemos visto cómo de erróneo estaba el resultado obtenido. Esto nos muestra que la implementación de los ordenadores cuánticos siguen en desarrollo, teniendo que esperar a estos para poder ver una implementación real de los algoritmos y que sea funcional.

Finalmente acabaremos con el breve resumen en inglés que se nos pide para el trabajo.

5.1. Abstract

As we said earlier, in this section we will write a brief abstract about the paper as a summary of the concepts learnt.

The quantum computing is the field of study that combines physics, mathematics and computer science. It exploits the quantum mechanical phenomena called quantum superposition in which a object is simultaneously at different states. With this phenomena, we can design a quantum sistem where the main object is the quantum bit or qubit. Instead of the basic states 0 and 1 of a bit, it can be in a superposition of the states $|0\rangle$ and $|1\rangle$.

This state of a qubit is represented by a vector in a vectorial space, so the superposition of states is just a lineal combination of the elements of a vectorial basis

$$|\varphi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle \tag{5.1}$$

We can measure this qubit to determine their state, in which the qubit collapse into one of the classical states 0 or 1.

We form bigger sistem of qubits by performing their tensor product as a vector to represent their states. We can also manipulate states with quantum gates, which functions as a classical logic gate, and we represent them with lineal aplications on the vectorial space. This comes with their corresponding matrix association, which only requirements is that they must be unitary, that is the product with their adjoint is the indentity matrix.

With all the tool, we can make basic quantum algorithm in which:

- We usually start by putting the qubits into superposition of many states.
- This is followed by acting on this qubits in superposition with some unitary operations.
- Then we measure the qubits to determine their state and reach some sort conclusion with the results.

There is more complex algorithm which involves more complex operations.

Referencias

[1] N. S. Yanofsky and M. A. Mannucci, *Quantum computing for computer scientists.* Cambridge University Press, 2008.