UNIVERSIDAD ABIERTA INTERAMERICANA
Facultad de Tecnología Informática

Carrera: Licenciatura en Matemática

Fundamentos Matemáticos de Computación Cuántica en el Algoritmo de Shor, para la factorización prima de números enteros.

Autor: Pablo Andrés Oviedo
Directora: Dra. Samira Abdel Masih

TESIS PRESENTADA PARA OPTAR AL TÍTULO DE LICENCIADO EN MATEMÁTICA

- Diciembre de 2017 -

Firmas del jurado

Agradecimientos

Al amor de mi vida, mi mujer, mi compañera, mi novia Verónica Pontieri, a quien le dedico especialmente este trabajo de tesis, por su amor y apoyo incondicional.
A la Dra. Samira Abdel Masih, por aceptar dirigir esta tesis, por su compromiso infinito con el trabajo y su máximo nivel de exigencia.
A mi familia completa, especialmente a mi madre Ana Maria, que me dio la vida, me enseñó a luchar y nunca bajar los brazos.
A mi padre que me protege desde el cielo, y ojala sienta orgullo del hombre que soy.
A mi abuela Isabel, que prendió una vela por cada examen que rendí.
A mi hermana Natalia y mis tres hermosos sobrinos, Agos, Lu y Cande.
A mis compañeros de cursada, que fueron piezas fundamentales en mi camino, de los cuales aprendí y aprendo día a día.
A mis docentes, en especial a Cristina, Lorena, Mabel, Soledad, Nicolas, Jorge, Samira, Nora, algunos de ellos también fueron mis compañeros, perdón a los que me olvide.
A la Universidad Abierta Interamericana, a sus autoridades, que me han formado como profesional y brindaron un contexto educativo de primer nivel.
A mis amigos, que me alentaron a seguir siempre para adelante.

Ni un paso atrás, siempre para adelante.

Infinitamente gracias!!!

Resumen

El presente trabajo desarrolla los fundamentos matemáticos básicos de Computación Cuántica, para luego analizar la primera parte de la publicación realizada por Peter Shor en 1995, conocida como "El Algoritmo de Shor" ([1] , págs. 1-10).
Dicho algoritmo, ideado para ser aplicado en una hipotética computadora cuántica, descompone en factores primos un número entero mayor que 1, en un tiempo considerablemente menor que el llevado a cabo por una computadora convencional.
Además, es exponencialmente más rápido que cualquier algoritmo clásico conocido.
Por ejemplo, a una computadora clásica le tomaría cientos de años poder encontrar los factores primos de un número de 600 dígitos, mientras que en una computadora cuántica sólo le tomaría unos minutos.
Pero, ¿por qué el interés de la factorización de un número entero, un tema que aparentemente está confinado sólo al Álgebra?
Porque el problema de la factorización es utilizado para codificar la mayoría de los mensajes secretos que se envían hoy en día.
Es decir, que el Algoritmo de Shor podría ser empleado para atacar las bases de seguridad en el almacenamiento y tránsito de la información. Esto reduciría drásticamente el nivel de complejidad que se necesita para descubrir numerosas claves de seguridad informática.
De este modo, muchas criptografías de clave pública, tales como el sistema RSA, que se aplica en las claves bancarias online, llegarían a ser obsoletas si el Algoritmo de Shor fuera implementado en una computadora cuántica.
La primera parte de la publicación de Shor, que se detalla en este trabajo, contiene las herramientas matemáticas e informáticas necesarias para implementar el algoritmo. En la segunda parte de la publicación se describe y demuestra el algoritmo, el cual será desarrollado por la tesista Yésica Valente.
A los efectos de facilitar al lector la comprensión de los temas, éstos se ordenan y exponen de la siguiente manera:

- Introducción: Se explica someramente en qué consiste una computadora cuántica.
- Capítulo 1: Se introducen las herramientas matemáticas básicas que se utilizarán en este trabajo.
- Capítulo 2: Se define el qubit, la unidad mínima de información cuántica. Se describen además sus propiedades.
- Capítulo 3: Se describen matemáticamente los circuitos y compuertas cuánticas.
- Capítulo 4: Se expone y analiza el Paralelismo Cuántico.
- Capítulo 5: Se demuestra un algoritmo cuántico para calcular la potencia modular.
- Capítulo 6: Se elaboran las conclusiones.

Palabras claves:

Algoritmo de Shor, qubits, computación cuántica, paralelismo cuántico, exponenciación modular.

Índice

Capítulo 6: Conclusiones

Introducción

¿Qué es una computadora cuántica?

Una computadora cuántica es un dispositivo que utiliza el modelo de los estados de ciertos elementos del átomo para realizar sus procesos.
Los ordenadores convencionales resumen toda la información que procesan a lenguaje binario, es decir, sólo utilizan dos estados para almacenar y operar con datos: **0** ó **1**.
La unidad mínima de información utilizada por estos ordenadores, para almacenar uno de esos dos valores, es el bit. Al igual que un interruptor, los bits sólo pueden estar encendidos o apagados. De este modo, toda la información en la computación actual se resume en una secuencia de ceros (bits apagados) y unos (bits encendidos).

Figura 0.1: Datos almacenados en una computadora convencional, como una secuencia de ceros y unos.
Fuente: http://computerhoy.com/noticias/hardware/que-es-computacion-cuantica-20079

En cambio, las partículas subatómicas (electrones, protones, neutrones, fotones, etc) tienen una curiosa cualidad que hace fascinante la computación cuántica. Esa cualidad es la **superposición**.
La **superposición** de una partícula subatómica consiste en que ésta no sólo puede adoptar el estado 0 ó 1, sino que puede estar en ambos estados al mismo tiempo.
Así, los ordenadores cuánticos son capaces de probar, al mismo tiempo, todas las posibilidades que existen para la solución concreta de un problema, en lugar de probarlas una tras otra, como lo realizan actualmente las computadoras convencionales.
En definitiva, las computadoras cuánticas procesan los datos en forma paralela, a diferencia de las computadoras actuales, que lo hacen en forma secuencial.

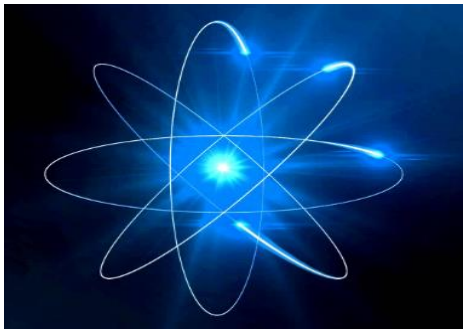


Figura 0.2: La superposición de las partículas subatómicas es la base de la computación cuántica.
Fuente: https://www.educacion-holistica.org/wordpress/

Este cambio en el paradigma de la computación supone un enorme salto hacia adelante, ya que permitirán realizar cálculos complejos que actualmente resultan inalcanzables en la computación clásica.

El qubit: la unidad mínima de información en una computadora cuántica.

Así como el bit es la unidad mínima de información en una computadora convencional, el bit cuántico o qubit es la unidad mínima de información en una computadora cuántica, y representa el estado de una partícula subatómica. Por lo general, y así se asumirá a lo largo de este trabajo, se considera el estado del electrón situado en la órbita más externa de un determinado átomo. De este modo, la información se codifica sobre el estado de dicho electrón. A mayor cantidad de qubits, más rápido funcionará un ordenador cuántico.

¿Cómo se codifica la información en una computadora cuántica?

Como se mencionó anteriormente, en un qubit se almacena el estado del electrón más externo de un átomo. Pero ¿cómo se determina el estado de un electrón?
Se lo determina en base a la propiedad de "**spín**" que tienen las partículas subatómicas.
El **spín** (que significa giro en inglés), descubierto en 1925, es la facultad que tienen dichas partículas de girar en torno a su eje. En particular, se considera el spín del electrón más externo de un átomo. Su sentido de giro queda determinado por "La Regla de la mano derecha". Si el electrón está **spín arriba**, se encuentra en su mayor nivel de energía y se supone que está en estado **0**. Mientras que si está **spín abajo**, su nivel energético es el más bajo y se dice que está en estado **1**.

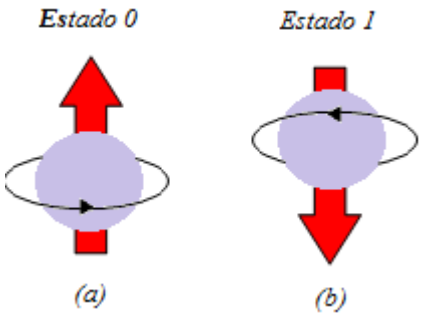


Figura 0.3: (a) Electrón girando spín arriba. En ese caso, se supone que hay almacenado un 0. (b) Electrón girando spin abajo. Se asume que hay almacenado un 1.

Pero el electrón también puede girar en cualquier otra posición comprendida entre el estado 0 y el estado 1. Es decir, puede rotar en una posición que es combinación lineal de ambos estados.

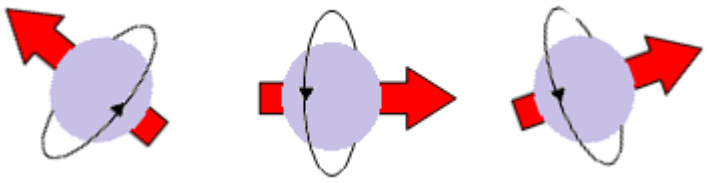


Figura 0.4: Distintas posiciones del eje de giro de un electrón

Para comprender este fenómeno, algunos comparan el spin del electrón con el movimiento de un trompo.

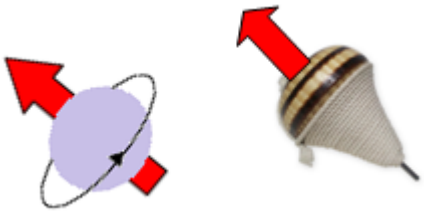


Figura 0.5: Analogía entre el giro de un electrón y el de un trompo.

Es por ello que, para medir la posición del eje de rotación del electrón en todas las posibles direcciones, se utiliza un par ordenado de números complejos. De este modo el qubit, que almacena la orientación del spin de un electrón, es representado mediante un vector complejo de dos componentes. Por ejemplo, si el electrón está spin arriba, el valor del qubit es $(1,0)^T$, y se lo simboliza por $| 0 >$. Mientras que, si está spin abajo, su valor es $(0,1)^T$ y se lo representa por $| 1 >$.
En caso de que el eje de rotación tome una posición distinta a las de spin arriba o spin abajo, se supone que el electrón está en ambos estados al mismo tiempo.
Por esta razón se afirma que un qubit puede adoptar simultáneamente los estados 0 y 1. Este hecho es lo que permite, a una computadora cuántica, procesar varios datos simultáneamente.

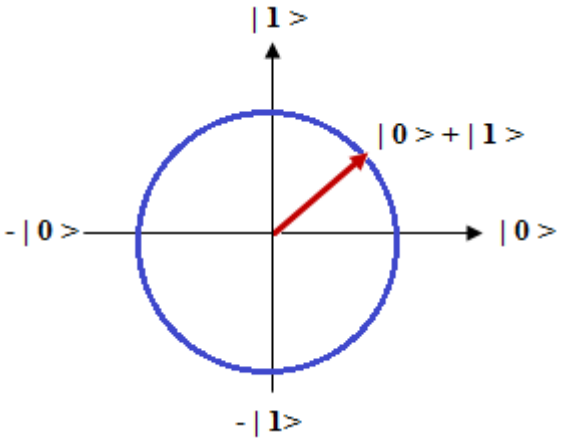


Figura 0.6: Representación vectorial de un qubit

En el Capítulo 2 se verá en más detalle cómo se describe matemáticamente un qubit, pero su concepto desde el punto de vista físico se resume a continuación:

Un qubit es la unidad mínima de información en una computadora cuántica. Representa el estado en que se encuentra el eje de giro o spin del electrón más externo de un átomo, el cual queda determinado mediante un par ordenado de números complejos. La información se codifica en base a los valores que toma el spin del electrón.

El entrelazamiento cuántico

Los inquietos y revoltosos átomos, además de que sus elementos pueden adoptar varios estados simultáneamente, también cuentan con otra propiedad llamada **entrelazamiento cuántico** (o **entanglement** en inglés). Gracias a esta particularidad, un electrón puede transmitir determinadas propiedades a otro sin que haya conexión física entre ellos. Esto hace que cualquier cambio en el estado de uno de los qubits entrelazados, provoque un cambio instantáneo en el otro. El entrelazamiento cuántico permitirá resolver en segundos tareas para las que una computadora convencional tardaría años.

En el Capítulo 2 también se verá cómo se traduce matemáticamente este concepto.

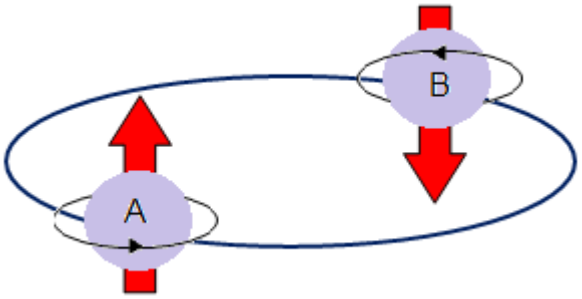


Figura 0.7: El entrelazamiento cuántico.

Este fenómeno produce que, si las partículas A y B están entrelazadas, al observar o medir el estado de giro de la partícula A, la partícula B recibirá automáticamente la "señal" y se mostrará girando, por ejemplo, en sentido contrario.

¿Cómo funciona una computadora cuántica?

En un procesador cuántico no se utilizan ni monitores, ni discos duros, ni ningún tipo de hardware tal como se conoce en el ámbito de la informática actual.
Todo sucede en la unidad de procesamiento que debe permanecer en unas condiciones de absoluto aislamiento, ya que los estados cuánticos del átomo son extremadamente frágiles y la superposición de los estados que se produce durante los procesos de cálculo, puede perturbarse.
Así, la superposición podría verse afectada ante el contacto con un campo electromagnético o la más mínima vibración o fluctuación de la temperatura.
Esta eventual alteración del estado cuántico del átomo provocaría errores de cálculo. Por esta razón, los procesadores en los que se encuentran los qubits deben enfriarse y mantenerse a cero absoluto (- 273 grados Celcius).
Los estados de cada qubit se observan mediante mediciones con láser, de las que se extraen los resultados de los cálculos y se procesan con ordenadores normales.

¿Existe en la actualidad una computadora cuántica?

El ordenador cuántico dejó de ser sólo una teoría para convertirse en realidad en el año 1998. Fue en ese entonces que el investigador Isaac Chuang dirigió a un grupo de la Universidad de Berkeley (California) para construir la primera computadora cuántica. Ésta funcionaba con sólo un qubit.
En 2001 un grupo de IBM desarrolló una computadora cuántica capaz de controlar 7 qubits. En ese año se probó por primera vez y de manera exitosa el Algoritmo de Shor, descomponiendo el número 15 en sus factores 3 y 5.
En 2013 se diseñó un ordenador cuántico que funcionaba con 512 qubits y, según aseguran sus desarrolladores, es hasta 108 milllones de veces más rápido que un ordenador tradicional.
Por el momento, empresas como Google, IBM y la NASA están implementando prototipos con la tecnología necesaria para lograr que, en unos 10 o 15 años, los ordenadores cuánticos sean una realidad cotidiana.

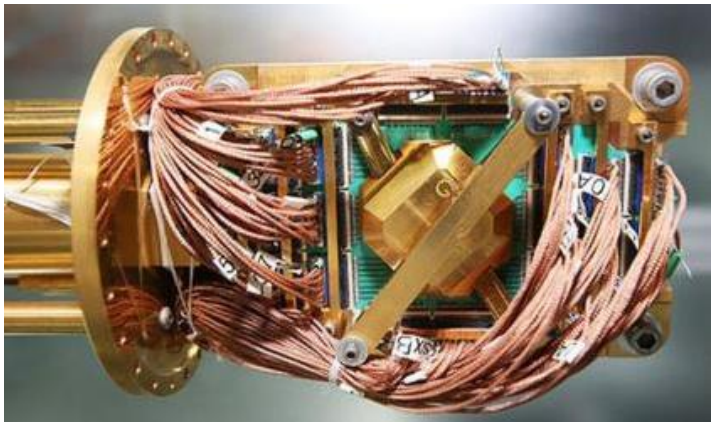


Figura 0.8: Imagen de un procesador cuántico.
Fuente: http://computerhoy.com/noticias/hardware/que-es-computacion-cuantica-20079

Dios no juega a los dados…

La computación Cuántica utiliza las herramientas de la Física y Mecánica Cuántica para implementar sus algoritmos cuánticos.
Pero ¿qué es la Física Cuántica? La Física Cuántica es una rama de la Física que estudia los fenómenos y propiedades que tienen las partículas subatómicas, en base a experiencias de laboratorio.
En cambio, la Mecánica Cuántica analiza en forma teórica, con formalismo puro, las propiedades y estados de dichas partículas.
Su desarrollo teórico se basa en seis leyes o postulados, llamados "Postulados de la Mecánica Cuántica". Éstos han sido deducidos mediante un largo proceso de prueba y error, y sus fundamentos siguen sorprendiendo, hasta el día de hoy, a múltiples autores.
Así por ejemplo, la Teoría de la Relatividad de Albert Einstein (1879-1955) establece que el Universo es ordenado y predecible. Pero en las leyes de la Mecánica Cuántica reina la incertidumbre. La única posibilidad para pronosticar algo es predecir todas las posibles soluciones o eventos que surgen de un hecho. En el mundo subatómico, todo es un juego de azar, como tirar los dados.
Albert Einstein rechazaba esta teoría y afirmaba que la Física no podría manejarse al azar, ya que sus leyes permiten predecir el resultado de un experimento. Por esta razón afirmaba que "***Dios no juega a los dados con el Universo…"***
Pero Niels Bohr (1886 – 1962), uno de los defensores y creadores de la Física Cuántica, le replicó diciendo: "***Albert, no le digas a Dios lo que tiene que hacer…***".

Capítulo 1

Nociones matemáticas básicas

Capítulo 1: Nociones matemáticas básicas

En este capítulo se expondrán los principales conceptos matemáticos que se utilizarán a lo largo de este trabajo. Primeramente se definirán los espacios de Hilbert, sobre los cuales se modelan matemáticamente los sistemas cuánticos. Luego se presentará el producto tensorial de vectores, que se aplicará para definir un sistema formado por varios qubits. Posteriormente se definirá y clasificará la complejidad de un algoritmo, lo cual permitirá demostrar la eficiencia del Algoritmo de Shor frente a otros algoritmos de factorización de números enteros. Finalmente se introducirán las nociones de Aritmética Modular y sus propiedades básicas.

1.1 Los espacios de Hilbert

El concepto de espacio de Hilbert, ideado por el matemático alemán David Hilbert (1862-1943), surge como una generalización del espacio Euclídeo. Esta generalización permite que ciertas nociones algebraicas y geométricas aplicables a espacios vectoriales de dimensión dos o tres, se extiendan a espacios de dimensión arbitraria e inclusive, de dimensión infinita (como el espacio vectorial formado por las funciones continuas).
Ejemplos de tales nociones son "la distancia entre vectores", "ortogonalidad de vectores" y "la proyección ortogonal".
Los espacios de Hilbert serán introducidos debido a que desempeñan un rol importante en la formulación matemática de la Mecánica Cuántica. Previamente se enunciarán los siguientes conceptos, que serán necesarios para comprender este tipo de espacios.

Definición 1.1: Producto interno

Sea **F** *el cuerpo de los números reales o de los complejos y* **V** *un espacio vectorial sobre* **F**. *Un "Producto interno" sobre* **V** *es una función que asigna, a cada par ordenado de vectores* (**v**, **w**), *un escalar* $< \mathbf{v}, \mathbf{w} >$ *de* **F** *de modo tal que* $\forall$ v, w, u $\in$ **V** *y* $\forall$ $\alpha \in$ **F** *, se verifican las siguientes propiedades:*

1) $< v + w, u > = < v , u > + < w , u >$
2) $< \alpha v , w > = \alpha < v , w >$
3) $< v , w > = \overline{< w, v >}$ *donde la barra indica conjugación compleja.*
4) $< v , v > \geq 0 \quad \forall \ v \in \mathbf{V}$
5) $< v, v > > 0$ *si* $v \neq \mathbf{0}$ *y donde* **0** *representa el vector nulo.*

La siguiente Proposición, que será útil para las demostraciones posteriores, es una consecuencia inmediata de las propiedades del producto interno:

Proposición 1.1

Si $v, w \in \mathbf{V}$ y $\alpha \in \mathbf{F}$ entonces $< v , \alpha\ w > = \overline{\alpha}\ < v , w >$

Demostración

Aplicando las Propiedades 3) y 4) del producto interno y la propiedad de la conjugación del producto de dos números complejos, resulta

$$< v , \alpha\ w > = \overline{< \alpha\ w, v >} = \overline{\alpha < w, v >} = \overline{\alpha}\ \overline{< w, v >} = \overline{\alpha} < v , w > \ \blacksquare$$

Observación 1.1

Si el cuerpo **F** es el de los números reales, es decir, si $\mathbf{F} = \mathbb{R}$, de la Propiedad 3) del producto interno se obtiene que $< v , w > = < w , v > \ \forall v, w \in \mathbf{V}$.

Pero si el cuerpo **F** es el de los números complejos, es decir, si $\mathbf{F} = \mathbb{C}$, la barra de conjugación que figura en la Propiedad 3) es absolutamente necesaria para evitar llegar a una contradicción en la definición de producto interno. Más específicamente, si se elimina la barra de conjugación en dicha Propiedad, resulta que

$$< v , w > = < w , v > \quad \forall v, w \in \mathbf{V}$$

En este caso, al aplicar también la Propiedad 2) se tiene

$$< v , \alpha\ v > = < \alpha\ v , v > = \alpha\ < v , v > \quad \forall v \in \mathbf{V} \text{ y } \forall\ \alpha \in \mathbf{F} \qquad [\ 1.1\ ]$$

Por otro lado, de la Propiedad 5) resulta que si $v \neq \mathbf{0}$ entonces $i\ v \neq \mathbf{0}$ y en consecuencia

$$< i\ v , i\ v > > 0 \qquad [\ 1.2\ ]$$

De las Propiedades 2) y 5) y la expresión [1.1] se obtiene que, si $v \neq \mathbf{0}$ entonces

$$< i\ v , i\ v > = i^2 < v , v > = -\ < v , v > < 0$$

Lo cual contradice la expresión [1.2] ■

A continuación se muestran algunos ejemplos de producto interno.

Ejemplo 1.1

1) Si $\mathbf{V} = \mathbf{F^n}$, se define el "producto interno canónico" de la siguiente manera:
Si $\mathbf{v, w} \in \mathbf{V}$ con $\mathbf{v} = (\mathbf{v_1, v_2, \ldots , v_n})^{\mathbf{T}}$ y $\mathbf{w} = (\mathbf{w_1, w_2, \ldots, w_n})^{\mathbf{T}}$ entonces

$$\mathbf{< v , w >} = \sum_{i=1}^{n} v_i\ \overline{w_i}$$

De este modo, cuando $\mathbf{F} = \mathbb{R}$ resulta

$$\mathbf{< v , w >} = \sum_{i=1}^{n} v_i\ w_i$$

En este caso, el producto interno canónico recibe el nombre de "producto escalar".

2) Sea $\mathbf{V} = \{ f : [a, b] \to \mathbb{C} \text{ tal que f es continua en } [a, b] \}$. Dadas $f, g \in \mathbf{V}$, si se define

$$< f , g > = \int_a^b f(x)\, \overline{g(x)}\, dx$$

Entonces se puede demostrar que es un producto interno sobre **V** ■

En todo espacio vectorial con producto interno queda definida la "norma o módulo de un vector" del siguiente modo:

Definición 1.2: Norma o módulo de un vector

Dado un espacio vectorial **V** *con producto interno, la norma o módulo de un vector* $v \in \mathbf{V}$ es

$$\| v \| = \sqrt{< v , v >}$$

Las Definiciones 1.1 y 1.2 dan lugar a presentar la definición de Espacio de Hilbert.

Definición 1.3: Espacio de Hilbert

Un espacio de Hilbert es un espacio vectorial **H** *definido sobre* $\mathbb{C}$ *que verifica las siguientes condiciones:*

1) *En* **H** *hay definido un producto interno.*
2) **H** *es un espacio vectorial completo. Esto significa que toda sucesión de Cauchy es convergente. Es decir,*

   Si $\{v_n\}_{n=1}^{\infty} \subset \mathbf{H}$ *y* $\| v_n - v_m \| \underset{n,\, m \to \infty}{\longrightarrow} 0$ *entonces* $\exists\, v \in \mathbf{H}$ *tal que*

   $\| v_n - v \| \underset{n \to \infty}{\longrightarrow} 0$

Ejemplo 1.2

1) $\mathbb{C}^n$, con el "producto interno canónico", es un espacio de Hilbert $\forall\, n \in \mathbb{N}$.

2) Si $\mathbf{V} = \{ f : [a, b] \to \mathbb{C} \text{ tal que f es continua en } [a, b] \}$ con el producto interno definido por $< f , g > = \int_a^b f(x)\, \overline{g(x)}\, dx$, se puede probar que **V** no es un espacio de Hilbert ■

De ahora en adelante, se trabajará con el espacio de Hilbert $\mathbb{C}^n$ y con el producto interno canónico.

Es decir, Si $\mathbf{v, w} \in \mathbb{C}^n$ con $\mathbf{v} = (v_1, v_2, \ldots, v_n)^T$ y $\mathbf{w} = (w_1, w_2, \ldots, w_n)^T$ entonces

$$< \mathbf{v} , \mathbf{w} > = \sum_{i=1}^{n} v_i \, \overline{w_i}$$

En este caso, la norma o módulo de un vector $\mathbf{v} = (v_1, v_2, \ldots, v_n)^T$ es

$$\| \mathbf{v} \| = \sqrt{\sum_{i=1}^{n} v_i \, \overline{v_i}} = \sqrt{\sum_{i=1}^{n} |v_i|^2}$$

En el próximo capítulo se mencionará la relación existente entre los espacios de Hilbert y la unidad mínima de información, que es el qubit.

1.2 Producto tensorial de vectores

Sean $\mathbf{V} = \mathbf{F^n}$ y $\mathbf{W} = \mathbf{F^m}$ dos espacios vectoriales de dimensión **n** y **m** respectivamente, con $\mathbf{F} = \mathbb{R}$ ó $\mathbb{C}$.

Si $\mathbf{v} = (v_1, v_2, \ldots, v_n)^T$ y $\mathbf{w} = (w_1, w_2, \ldots, w_m)^T$ entonces el producto tensorial de **v** con **w** , simbolizado por $\mathbf{v} \otimes \mathbf{w}$, se lo define así:

$$v \otimes w = \begin{pmatrix} v_1 \, w \\ v_2 \, w \\ . \\ . \\ v_n \, w \end{pmatrix} = \begin{pmatrix} v_1 \begin{pmatrix} w_1 \\ w_2 \\ . \\ w_m \end{pmatrix} \\ v_2 \begin{pmatrix} w_1 \\ w_2 \\ . \\ w_m \end{pmatrix} \\ . \\ . \\ v_n \begin{pmatrix} w_1 \\ w_2 \\ . \\ w_m \end{pmatrix} \end{pmatrix} = \begin{pmatrix} \begin{pmatrix} v_1 \, w_1 \\ v_1 w_2 \\ . \\ v_1 \, w_m \end{pmatrix} \\ \begin{pmatrix} v_2 \, w_1 \\ v_2 \, w_2 \\ . \\ v_2 \, w_m \end{pmatrix} \\ . \\ . \\ \begin{pmatrix} v_n \, w_1 \\ v_n \, w_2 \\ . \\ v_n \, w_m \end{pmatrix} \end{pmatrix}$$

Claramente se puede verificar que el producto tensorial no es conmutativo. Es decir,

$$\mathbf{v} \otimes \mathbf{w} \neq \mathbf{w} \otimes \mathbf{v}$$

Las propiedades básicas del producto tensorial se enumeran en la siguiente Proposición:

Proposición 1.2

1) *Si* $\mathbf{u} \in \mathbf{F^n}$ y $\mathbf{v}, \mathbf{w} \in \mathbf{F^m}$ *entonces* $\mathbf{u} \otimes (\mathbf{v} + \mathbf{w}) = (\mathbf{u} \otimes \mathbf{v}) + (\mathbf{u} \otimes \mathbf{w})$

2) *Si* $\mathbf{u}, \mathbf{v} \in \mathbf{F^n}$ y $\mathbf{w} \in \mathbf{F^m}$ *entonces* $(\mathbf{u} + \mathbf{v}) \otimes \mathbf{w} = (\mathbf{u} \otimes \mathbf{w}) + (\mathbf{v} \otimes \mathbf{w})$

3) *Si* $\mathbf{u} \in \mathbf{F^n}$, $\mathbf{v} \in \mathbf{F^m}$ y $\alpha \in \mathbb{C}$ *entonces*

$$(\alpha\,\mathbf{u}) \otimes \mathbf{v} = \alpha\,(\mathbf{u} \otimes \mathbf{v})$$
$$\mathbf{u} \otimes (\alpha\,\mathbf{v}) = \alpha\,(\mathbf{u} \otimes \mathbf{v})$$

4) Si *Si* $\mathbf{u} \in \mathbf{F^n}$, $\mathbf{v} \in \mathbf{F^m}$ y $\alpha > \mathbf{0}$ *entonces* $\alpha\,(\mathbf{u} \otimes \mathbf{v}) = (\sqrt{\alpha}\,\mathbf{u}) \otimes (\sqrt{\alpha}\,\mathbf{v})$

Demostración

1) Si $u = (u_1, u_2, \ldots, u_n)^T$, $v = (v_1, v_2, \ldots, v_m)^T$ y $w = (w_1, w_2, \ldots, w_m)^T$ entonces

$$u \otimes (v + w) = \begin{pmatrix} u_1 \\ u_2 \\ . \\ . \\ u_n \end{pmatrix} \otimes \left[\begin{pmatrix} v_1 \\ v_2 \\ . \\ . \\ v_m \end{pmatrix} + \begin{pmatrix} w_1 \\ w_2 \\ . \\ . \\ w_m \end{pmatrix} \right] = \begin{pmatrix} u_1 \\ u_2 \\ . \\ . \\ u_n \end{pmatrix} \otimes \begin{pmatrix} v_1 + w_1 \\ v_2 + w_2 \\ . \\ . \\ v_m + w_m \end{pmatrix} =$$

$$= \begin{pmatrix} u_1 . \begin{pmatrix} v_1 + w_1 \\ v_2 + w_2 \\ . \\ v_m + w_m \end{pmatrix} \\ . \\ . \\ . \\ u_n . \begin{pmatrix} v_1 + w_1 \\ v_2 + w_2 \\ . \\ . \\ v_m + w_m \end{pmatrix} \end{pmatrix} = \begin{pmatrix} u_1 . \begin{pmatrix} v_1 \\ v_2 \\ . \\ v_m \end{pmatrix} + u_1 . \begin{pmatrix} w_1 \\ w_2 \\ . \\ w_m \end{pmatrix} \\ . \\ . \\ . \\ u_n . \begin{pmatrix} v_1 \\ v_2 \\ . \\ . \\ v_m \end{pmatrix} + u_n . \begin{pmatrix} w_1 \\ w_2 \\ . \\ w_m \end{pmatrix} \end{pmatrix} = \begin{pmatrix} u_1 . \begin{pmatrix} v_1 \\ v_2 \\ . \\ v_m \end{pmatrix} \\ . \\ . \\ . \\ u_n . \begin{pmatrix} v_1 \\ v_2 \\ . \\ . \\ v_m \end{pmatrix} \end{pmatrix} + \begin{pmatrix} u_1 . \begin{pmatrix} w_1 \\ w_2 \\ . \\ w_m \end{pmatrix} \\ . \\ . \\ . \\ u_n . \begin{pmatrix} w_1 \\ w_2 \\ . \\ . \\ w_m \end{pmatrix} \end{pmatrix} =$$

$$= (u \otimes v) + (u \otimes w).$$

Los incisos 2), 3) y 4) se demuestran de manera similar. ■

Observación 1.2

A fin de simplificar la notación, se eliminarán los paréntesis internos que figuran en el resultado de un producto tensorial.

Es decir, si $\mathbf{v} = (v_1, v_2, \ldots, v_n)^T$ y $\mathbf{w} = (w_1, w_2, \ldots, w_m)^T$ entonces

$$v \otimes w = \begin{pmatrix} \begin{pmatrix} v_1 w_1 \\ v_1 w_2 \\ . \\ v_1 w_m \end{pmatrix} \\ \begin{pmatrix} v_2 w_1 \\ v_2 w_2 \\ . \\ v_2 w_m \end{pmatrix} \\ . \\ . \\ \begin{pmatrix} v_n w_1 \\ v_n w_2 \\ . \\ v_n w_m \end{pmatrix} \end{pmatrix} \approx \begin{pmatrix} v_1 w_1 \\ v_1 w_2 \\ . \\ v_1 w_m \\ v_2 w_1 \\ v_2 w_2 \\ . \\ v_2 w_m \\ . \\ . \\ v_n w_1 \\ v_n w_2 \\ . \\ v_n w_m \end{pmatrix}$$

De este modo, si $v \in F^n$ y $w \in F^m$ el producto tensorial $v \otimes w$ será identificado mediante un vector en $F^{n\,m}$.

1.3 Orden de complejidad computacional de un algoritmo

La resolución práctica de un problema exige por una parte un algoritmo o método de resolución y por otra parte, para que pueda ser ejecutado por una computadora, su programación o codificación.

Desde el punto de vista de la Ingeniería y de la Informática, lo que preocupa son los recursos físicos necesarios para que un algoritmo pueda ejecutarse. De todos ellos, los más importantes son dos: **el tiempo de ejecución** y **la cantidad de memoria** que requiere el algoritmo. Estos dos parámetros definen básicamente el orden de complejidad computacional de un algoritmo y, para medirlos, se utiliza el concepto de "funciones de orden O". Su definición se da a continuación:

Definición 1.4: Funciones de orden O

Sea f: $\mathbb{N} \to [0, +\infty)$.*El conjunto de funciones de orden* O *de* f, *simbolizado por* O(f) *es*

$$O(f) = \{\, g: \mathbb{N} \to [0, +\infty) \text{ tal que } \exists\, k > 0 \ \wedge\ n_0 \in \mathbb{N} \,/\, g(n) \le k\, f(n)\ \forall\ n \ge n_0 \,\}$$

Así, una función g: $\mathbb{N} \to [0, +\infty)$ se dice que es O(f) si $g \in O(f)$.

Ejemplo 1.3

Si $g(n) = 2n^3 + 4n + n^3 \operatorname{sen}(n)$ y $f(n) = n^3$, probar que g es O(f).

Solución

Se observa que, $\forall\ n \geq 1$ se verifica que

$$\frac{g(n)}{f(n)} = \frac{2n^3 + 4n + n^3\operatorname{sen}(n)}{n^3} = \frac{2n^3}{n^3} + \frac{4n}{n^3} + \frac{n^3\operatorname{sen}(n)}{n^3} = 2 + \frac{4}{n^2} + \operatorname{sen}(n) \leq 2 + 4 + 1 = 7$$

Por lo tanto, tomando $n_0 = 1$ y $k = 7$ se verifica que $g(n) \leq k\, f(n)\ \forall\ n \geq n_0$. En consecuencia, $g \in O(f)$ ■

La siguiente Proposición será útil para algunas demostraciones que se desarrollarán en el Capítulo 5.

Proposición 1.3

1) *Si* $\lim\limits_{n\to\infty} \frac{g(n)}{f(n)} \leq M$ *para alguna constante* M *entonces* $g \in O(f)$.

2) *Si* $f_1 \in O(g)$ y $f_2 \in O(h) \Rightarrow f_1 . f_2 \in O(g.h)$

Demostración

1) Sea $L = \lim\limits_{n\to\infty} \frac{g(n)}{f(n)}$. Entonces, por definición de límite, se tiene que

$\forall\ \varepsilon > 0\ \exists\ n_0 / \forall\ n \geq n_0$ se verifica que $\left| \frac{g(n)}{f(n)} - L \right| < \varepsilon$.

En consecuencia, $\forall\ n \geq n_0$ resulta

$\frac{g(n)}{f(n)} - L < \varepsilon \Rightarrow \frac{g(n)}{f(n)} < \varepsilon + L < \varepsilon + M$. Luego, tomando $k = \varepsilon + M$ se obtiene que

$\forall\ n \geq n_0\ \ \frac{g(n)}{f(n)} < k \Rightarrow g(n) < k\, f(n)$. Luego, $g \in O(f)$.

2) Como $f_1 \in O(g)$ y $f_2 \in O(h)$ resulta

$\exists\ k_1 > 0 \wedge n_1 \in \mathbb{N}\ /\ f_1(n) \leq k_1\ g(n)\ \forall\ n \geq n_1$ y además

$\exists\ k_2 > 0 \wedge n_2 \in \mathbb{N}\ /\ f_2(n) \leq k_2\ h(n)\ \forall\ n \geq n_2$

Tomando $k = k_1 k_2$ y $n_0 = \max\{n_1, n_2\}$ entonces $\forall\ n \geq n_0$ se verifica que

$\frac{f_1(n)\, f_2(n)}{g(n)\, h(n)} = \frac{f_1(n)}{g(n)}\ \frac{f_2(n)}{h(n)} \leq k_1\, k_2 = k$. Por lo tanto, $f_1 . f_2 \in O(g.h)$ ■

Observación 1.3

A menudo, para enfatizar que las funciones f y g tienen por variable independiente a un número natural n, en lugar de afirmar que g es O(f) se suele decir que g(n) es O(f(n)). Esta notación es la que se aplicará en adelante.

A continuación se definirá el orden de complejidad computacional de un algoritmo, concepto que será aplicado en el Capítulo 5 para comparar la eficiencia de los nuevos algoritmos cuánticos que se plantean.

Definición 1.5: Orden de complejidad computacional de un algoritmo

Se dice el orden de complejidad de un algoritmo es O(f(n)) *donde* f: $\mathbb{N} \rightarrow [0, +\infty)$ *si el tiempo de ejecución o la cantidad de memoria requerida para ejecutar el algoritmo se expresa mediante una función* g(n), *con* g(n) $\in$ O(f(n))

En este caso, "n" puede representar la longitud de un vector, el orden de una matriz, el número de registros de una base de datos, etc. Por otro lado, la definición de O(f) permite asegurar que, a partir de cierto "n" en adelante, el tiempo de ejecución de un algoritmo o la cantidad de memoria requerida nunca será superior a f(n). Es decir que, si "n" es suficientemente grande, y salvo una constante multiplicativa "k", f(n) es una cota superior de g(n). Además, el hecho de indicar, en la definición de O(f), que "n" sea mayor que cierto valor n_0 se debe a que la eficiencia de un algoritmo se mide para valores de "n" muy grandes. Ya que, casi siempre, los problemas pequeños se pueden resolver de cualquier manera, apareciendo las limitaciones al atacar problemas grandes.
Por lo general, la función f(n) se escoge de modo tal que sea lo más sencilla posible. Esto da lugar a la siguiente definición.

Definición 1.6: Clasificación del orden de complejidad de un algoritmo

1) *Si* f(n) = log(n) *se dice que el algoritmo es de orden logarítmico.*

2) Si f(n) = n^p *con* p $\in \mathbb{N}$ *se dice que el algoritmo es de orden polinomial.*
En este caso, se lo considera ***eficiente****.*

3) *Si* f(n) = a^n *con* a > 1, *se dice que el algoritmo es de orden exponencial o No Polinomial (NP).*
En este caso, se lo considera ***ineficiente****.*

Los algoritmos de **orden logarítmico** son los ideales, ya que el tiempo que demandan para procesar los datos se reduce considerablemente.
Los de **orden polinomial** se enfrentan a una dificultad a medida que "p" crece. Mientras menor sea el valor de "p", más eficiente será el algoritmo. Complejidades del orden $O(n^2)$ y $O(n^3)$ suelen ser efectivamente abordables, y prácticamente nadie acepta algoritmos de orden $O(n^{100})$.
Los algoritmos de **orden exponencial** suelen ser intratables y sólo son aplicables a problemas con valores de "n" pequeños.
A la vista de lo expuesto, se comprende que los programadores busquen algoritmos de orden polinomial, ya que es un golpe de suerte encontrar de orden logarítmico.
Se finalizará este capítulo mostrando un ejemplo que ilustrará cómo clasificar el orden de complejidad de un algoritmo.

Ejemplo 1.4

Sean $\mathbf{v} = (v_1, v_2, ..., v_n)^T$ y $\mathbf{w} = (w_1, w_2, ..., w_n)^T$ dos vectores en $\Re^n$. Se tiene un ordenador que tarda T unidades de tiempo para efectuar una operación aritmética (suma, resta, multiplicación o división). Probar que el tiempo requerido para efectuar el producto escalar de **v** con **w** es O(n).

Solución

El producto escalar de **v** con **w** es

$$< \mathbf{v} , \mathbf{w} > = \sum_{i=1}^{n} v_i \, w_i$$

Que es el algoritmo cuya complejidad computacional se pretende analizar.
Para efectuar este cálculo se necesitan n multiplicaciones y (n-1) sumas. Por lo tanto, son necesarias n + (n-1) = 2 n -1 operaciones aritméticas. Luego, el ordenador tardará (2 n – 1) T unidades de tiempo para efectuar el producto escalar.
En este caso, g(n) = (2 n – 1) T. Si se define f(n) = n entonces, del inciso 1) de la Proposición 1.3 resulta

$$\lim_{n\to\infty} \frac{g(n)}{f(n)} = \lim_{n\to\infty} \frac{(2\,n-1)\,T}{n} = 2\,T$$

Tomando M = 2 T se obtiene que $g \in O(n)$ y en consecuencia, el tiempo requerido para efectuar el producto escalar es O(n) ■

1.4 Nociones de Aritmética Modular

Definición 1.7: Congruencia

Dado un número natural $n > 1$ y dos números enteros a y b, se dice que a es congruente con b módulo n, y se escribe

$$a \equiv b \pmod{n}$$

Si

$$n \mid (a - b)$$

La siguiente Proposición, cuya demostración se puede encontrar en cualquier bibliografía de Aritmética Modular, será aplicada en el capítulo referente a Exponenciación Modular:

Proposición 1.4

1) $a \equiv b \pmod{n} \Leftrightarrow$ a y b tienen el mismo resto en la división entera por n.
2) $a \equiv 0 \pmod{n} \Leftrightarrow n \mid a$
3) Si $a \equiv b \pmod{n} \wedge c \equiv d \pmod{n} \Rightarrow a\,c \equiv b\,d \pmod{n}$
4) Si $a \equiv b \pmod{n} \wedge c \equiv d \pmod{n} \Rightarrow a + c \equiv b + d \pmod{n}$
5) $a \equiv b \pmod{n} \wedge c \neq 0 \Rightarrow a\,c \equiv b\,c \pmod{n}$
6) $a \equiv b \pmod{n} \wedge k \in \mathbb{N} \Rightarrow a^k \equiv b^k \pmod{n}$
7) $a \equiv r_a \pmod{n}$ donde r_a es el resto de la división entera de a por n.

Definición 1.8: Inverso multiplicativo modular

Dado $a \in \mathbb{Z}$ y $n \in \mathbb{N}$, se dice que a tiene inverso multiplicativo (mod n) si $\exists\, b \in \mathbb{Z}$ tal que $a\,b \equiv 1 \pmod{n}$

Notación: Al número b, el inverso multiplicativo de a, se lo simbolizará por a^{-1}.

El siguiente Teorema indicará en qué casos existe el inverso multiplicativo modular.

Teorema 1.1

Dado $a \in \mathbb{Z}$ y $n \in \mathbb{N}$, a tiene inverso multiplicativo (mod n) $\Leftrightarrow$ mcd (a, n) = 1
Donde mcd (a, n) es el máximo común divisor entre a y n.

Demostración

Será desarrollada por la tesista Yésica Valente.

Capítulo 2

Los Qubits

Capítulo 2: Los Qubits

La unidad mínima y por lo tanto constitutiva de la teoría de la Informática Cuántica es el qubit, el análogo del bit en la Informática Clásica.
En el capítulo introductorio se definió el qubit desde el punto de vista físico, pero en este capítulo se lo definirá desde el enfoque matemático. Se mostrará su representación geométrica mediante la Esfera de Bloch y, además, se indicará cómo medirlo.
Posteriormente se introducirá y desarrollará el sistema cuántico formado por dos o más qubits junto con sus propiedades fundamentales: la superposición y el entrelazamiento.

2.1 Sistemas formados por un qubit

Los sistemas computacionales cuánticos formados por un qubit procesan la información utilizando un qubit. Su definición se expone a continuación.

Definición 2.1: qubit

Un qubit es un vector $\mathbf{q} \in \mathbb{C}^2$ *de módulo 1. Es decir,*

$$\mathbf{q} = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$$

donde $\alpha , \beta \in \mathbb{C}$ *y verifican que* $|\alpha|^2 + |\beta|^2 = 1$

La palabra **qubit** proviene de la unión de los vocablos ingleses **qu**antum **bit**. Se le dio ese nombre a fin de establecer analogías con el **bit** de la computación clásica.
Como un qubit se puede escribir de la forma

$$\mathbf{q} = \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \alpha \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \beta \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Se dice que el conjunto $\left\{ \begin{pmatrix} 1 \\ 0 \end{pmatrix} , \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\}$ es una base ortonormal para el sistema formado por un qubit. Por otro lado, si se establece la siguiente notación:

$$|0> = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad ; \quad |1> = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Entonces la Definición 2.1 se puede reformular de la siguiente manera:

Definición 2.2:

Un qubit es un vector **q** *de la forma*

$$\mathbf{q} = \alpha\ |\mathbf{0}> + \beta\ |\mathbf{1}>$$

Donde

$|0> = \begin{pmatrix}1\\0\end{pmatrix}$; $|1> = \begin{pmatrix}0\\1\end{pmatrix}$ y $\alpha, \beta \in \mathbb{C}$ *con* $|\alpha|^2 + |\beta|^2 = 1$

En este caso, el conjunto $\{ |0>, |1> \}$ *constituye una base para el sistema formado por un qubit.*

Los vectores $|0>$ y $|1>$ se corresponden con el 0 y 1 del bit clásico.
Si $q = |0>$ se dice que el qubit está en estado 0, mientras que si $q = |1>$ el qubit está en estado 1.
Cuando α y β no son nulos, se dice que el qubit está en estado de superposición, lo cual significa que se encuentra simultáneamente en los estados clásicos 1 y 0.

¿Cómo se modelan matemáticamente los sistemas formados por un qubit?

A fin de dar un marco teórico a este tipo de sistemas y analizar su evolución a lo largo del tiempo, éstos se modelan matemáticamente. Para ello se recurre a los Postulados de la Mecánica Cuántica y en particular, al primero de ellos:

Primer Postulado de la Mecánica Cuántica:

Asociado a cada ***sistema físico****, se encuentra un espacio de Hilbert* **H** *, conocido como* ***espacio de estados*** *del sistema. El sistema es completamente descripto por su* ***vector de estado****, el cual es un* ***vector unitario*** *en dicho espacio de estados.*

No es el propósito de este trabajo desarrollar esta teoría cuántica, sino simplemente mencionarla someramente:

El sistema físico al que hace referencia el Primer Postulado es el conjunto formado por una o más partículas subatómicas. Para el caso que se está analizando, el sistema físico está constituido por el electrón de la órbita más externa de un átomo. Los estados de un sistema son aquellas magnitudes físicamente medibles, como por ejemplo la velocidad, la aceleración, el momento angular, la energía, etc.
En este caso, el estado que se estudia de este sistema es la posición del eje de giro o spin del electrón.
Pero ¿Cómo se define matemáticamente el espacio de Hilbert asociado a este sistema? Para responder a esta pregunta cabe recordar que las partículas subatómicas, además de comportarse como partículas, se comportan también como ondas que se propagan por el espacio. Por esta razón, el estado de un sistema suele describirse mediante una función de onda $\psi(\mathbf{x},\mathbf{t})$, que es una función a valores complejos y cuyas variables independientes son la posición **x** y el tiempo **t**.

Para el caso que se está abordando, ψ **(x, t)** representa el estado del spin del electrón cuando éste se encuentra en la posición **x** y en el instante **t**. De este modo, el espacio de Hilbert asociado al sistema físico es el conjunto de todas las funciones de onda ψ **(x, t).** El vector de estado en este caso es el qubit que, por pertencer al espacio de Hilbert, deberá expresarse en términos de una función de onda.
Es decir,

$$\psi(\mathbf{x}, \mathbf{t}) = \mathbf{q} = \alpha \ |\mathbf{0}\rangle + \beta \ |\mathbf{1}\rangle$$

Con $\psi(\mathbf{x}, \mathbf{t})$ tal que $|\psi(\mathbf{x}, \mathbf{t})| = \mathbf{1}$.
Además, de acuerdo al enfoque que se dará a este tema, **q** no depende de **x** ni de **t**.
Éste fue el motivo por el cual Paul Dirac (1902-1984) representó al qubit mediante la siguiente notación:

$$|\psi\rangle = \alpha \ |\mathbf{0}\rangle + \beta \ |\mathbf{1}\rangle$$

En este trabajo no se analizará el comportamiento ondulatorio del estado de un sistema cuántico, sino que será desarrollado en términos vectoriales y matriciales. De este modo, como un qubit es representado mediante un vector en $\mathbb{C}^2$, se identificará al espacio de Hilbert asociado a este sistema físico con $\mathbb{C}^2$ ■

Ahora, cabe preguntarse: ¿cómo puede el qubit, un vector complejo de dos componentes, guardar la información del spin del electrón? En la próxima sección se verá que las componentes de dicho vector complejo determinan la dirección y sentido del eje de rotación del electrón. Esto se logra si a cada qubit se le hace corresponder un punto de una esfera de radio 1, llamada la "Esfera de Bloch".

Representación geométrica de un qubit: La Esfera de Bloch

En Mecánica Cuántica, la Esfera de Bloch es una esfera centrada en el origen y de radio uno, utilizada para representar geométricamente a los qubits. Su nombre alude al físico suizo Félix Bloch (1905-1983). Bloch tuvo la idea de asignar, a cada qubit, un único punto de dicha esfera, el cual permitirá visualizar el estado de dicho qubit, es decir, la posición del eje de rotación del electrón.

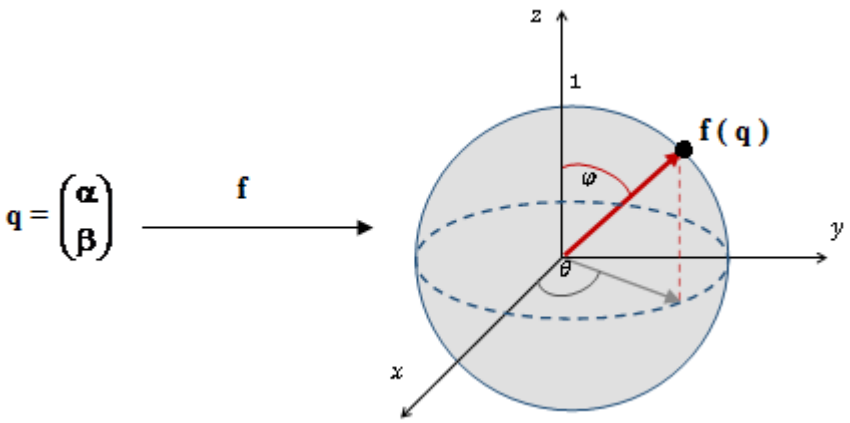


Figura 2.1: Representación de un qubit en la Esfera de Bloch.
Mediante una determinada función f, a cada qubit se le asigna un único punto de dicha esfera.

Para poder establecer esta correspondencia, se recordarán los siguientes conceptos:

Forma exponencial de un número complejo

Dado un numero complejo z = a + i b con a, b $\in \Re$, su representación en forma exponencial es

$$z = \rho\, e^{i\,\theta}$$

Donde

$$\begin{cases} \rho = \sqrt{a^2 + b^2} & (\rho \geq 0) \\ \theta = \text{arctg}\left(\dfrac{b}{a}\right) & \text{con } 0 \leq \theta < 2\,\pi \end{cases}$$

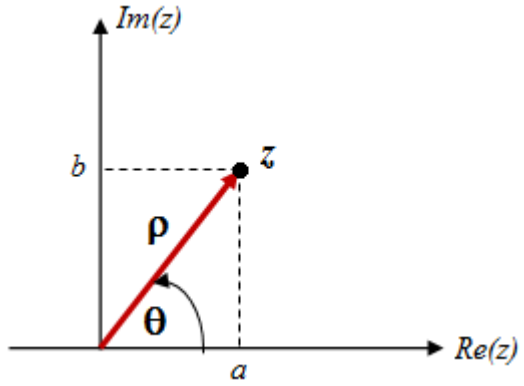


Figura 2.2: Representación de un número complejo en forma exponencial

Observación 2.1

Si z = 0 entonces a = b = 0. Al reemplazar estos valores en la expresión de θ se obtiene

$$\theta = \text{arctg}\left(\frac{b}{a}\right) = \text{arctg}\left(\frac{0}{0}\right)$$

Lo cual resulta una indeterminación. En este caso se supone que θ puede tomar cualquier valor. En este trabajo se considerará que, si z = 0 entonces θ = 0.

Coordenadas esféricas

Las coordenadas esféricas permiten identificar un punto en el espacio. En este sistema, un punto P = (x, y , z) se representa por medio de una terna ordenada (ρ, θ, φ) donde:

ρ : Distancia de P al origen (ρ ≥ 0)
θ : Ángulo formado por el semieje x positivo y la semirrecta que une el origen con el punto (x , y) (0 ≤ θ < 2 π)
φ : Ángulo formado por el semieje z positivo y la semirrecta que une el origen con el punto P = (x , y , z) (0 ≤ φ ≤ π)

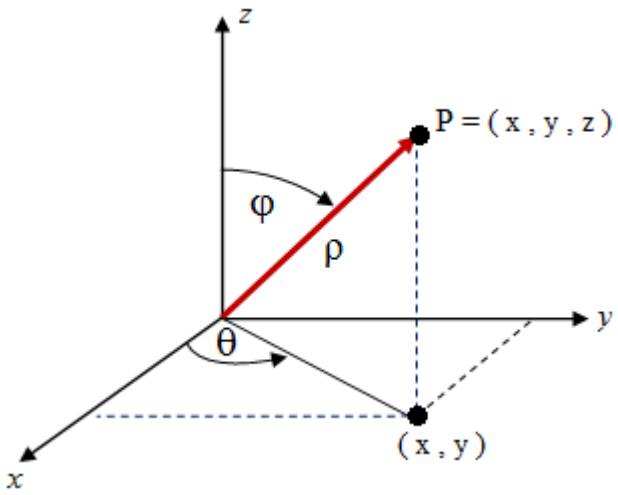


Figura 2.3: Coordenadas esféricas

Se puede probar que las coordenadas del punto P = (x , y , z) expresadas en coordenadas esféricas son:

$$\begin{cases} x = \rho \operatorname{sen}(\varphi) \cos(\theta) \\ y = \rho \operatorname{sen}(\varphi) \operatorname{sen}(\theta) \\ z = \rho \cos(\varphi) \end{cases}$$

Por lo tanto, si un punto P = (x , y , z) pertenece a la esfera centrada en el origen y de radio 1, cuya ecuación cartesiana es

$$x^2 + y^2 + z^2 = 1$$

Entonces se deduce que $\rho = 1$ y en consecuencia resulta

$$P = \left(\operatorname{sen}(\varphi) \cos(\theta) , \operatorname{sen}(\varphi) \operatorname{sen}(\theta) , \cos(\varphi)\right)$$

La siguiente Proposición indica de qué manera se asigna, a cada qubit, un punto de la Esfera de Bloch.

Proposición 2.1

Sea $q = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$ *un qubit y* $\begin{cases} \alpha = \rho_1 e^{i\theta_1} \\ \beta = \rho_2 e^{i\theta_2} \end{cases}$ *las representaciones de* α *y* β *en forma exponencial. Entonces la función*

$$f(q) = \left(\operatorname{sen}(\varphi) \cos(\theta) , \operatorname{sen}(\varphi) \operatorname{sen}(\theta) , \cos(\varphi)\right)$$

Donde

$$\varphi = 2 \operatorname{arctg}\left(\frac{\rho_2}{\rho_1}\right) \quad y \quad \theta = \begin{cases} \theta_2 - \theta_1 & \text{si} \quad 0 \le \theta_2 - \theta_1 < 2\pi \\ \theta_2 - \theta_1 + 2\pi & \text{si} \quad -2\pi < \theta_2 - \theta_1 < 0 \end{cases}$$

Asigna a cada qubit un único punto en la esfera de Bloch. Además, la función f es suryectiva.

Demostración

Primeramente se definirán los ángulos θ y φ con $0 \leq \theta < 2\pi$ y $0 \leq \varphi \leq \pi$ de modo tal que a cada qubit $q = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$ le quede asignado una terna $(1, \theta, \varphi)$ que representará un punto, en coordenadas esféricas, de la esfera de Bloch.

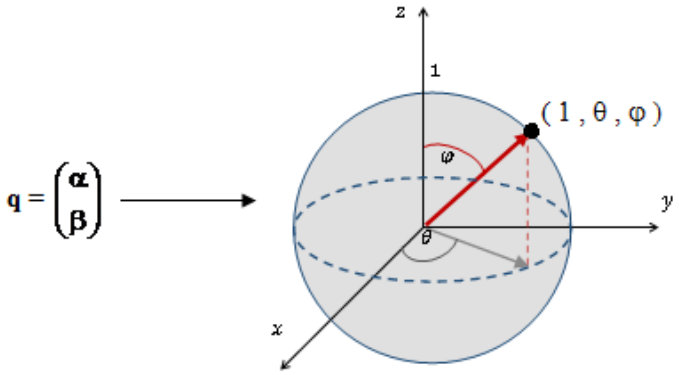


Figura 2.4: Asignación, a cada qubit ,de una terna $(1, \theta, \varphi)$ *en la esfera de Bloch.*

Si se expresa el par ordenado (ρ_1, ρ_2) en coordenadas polares, resulta

$$\begin{cases} \rho_1 = r\cos(\varphi_1) \\ \rho_2 = r\,\mathrm{sen}(\varphi_1) \end{cases}$$

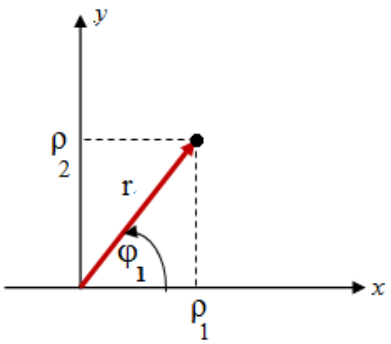


Figura 2.5: Representación de (ρ_1, ρ_2) *en coordenadas polares*

Como

$$|\alpha|^2 + |\beta|^2 = 1 \Rightarrow \left|\rho_1 e^{i\theta_1}\right|^2 + \left|\rho_2 e^{i\theta_2}\right|^2 = 1 \Rightarrow (\rho_1)^2 + (\rho_2)^2 = 1$$

$$\Rightarrow \left(r\cos(\varphi_1)\right)^2 + \left(r\,\mathrm{sen}(\varphi_1)\right)^2 = 1 \Rightarrow r^2 = 1 \Rightarrow r = 1$$

Por lo tanto,

$$q = \begin{pmatrix} \cos(\varphi_1)\, e^{i\theta_1} \\ \mathrm{sen}(\varphi_1)\, e^{i\theta_2} \end{pmatrix}$$

Por otro lado, debido a que $\rho_1, \rho_2 \geq 0$ se deduce que $0 \leq \varphi_1 \leq \frac{\pi}{2}$.

Si se divide **q** por $e^{i\theta_1}$ se obtiene

$$\frac{q}{e^{i\,\theta_1}} = \begin{pmatrix} \cos(\varphi_1) \\ \text{sen}(\varphi_1)\; e^{i\,(\theta_2-\theta_1)} \end{pmatrix}$$

Como $0 \le \varphi_1 \le \frac{\pi}{2}$ y $\varphi_1 = \text{arctg}\left(\frac{\rho_2}{\rho_1}\right)$ entonces $0 \le \text{arctg}\left(\frac{\rho_2}{\rho_1}\right) \le \frac{\pi}{2}$.

Luego, si se define

$$\varphi = 2\ \text{arctg}\left(\frac{\rho_2}{\rho_1}\right)$$

Resulta $0 \le \varphi \le \pi$.

Además, de las desigualdades $0 \le \theta_1 , \theta_2 < 2\,\pi$ resulta que $-2\,\pi < \theta_2 - \theta_1 < 2\,\pi$

En consecuencia, si se define

$$\theta = \begin{cases} \theta_2 - \theta_1 & \text{si} \quad 0 \le \theta_2 - \theta_1 < 2\,\pi \\ \theta_2 - \theta_1 + 2\,\pi & \text{si} \quad -2\,\pi < \theta_2 - \theta_1 < 0 \end{cases}$$

Se obtiene que $0 \le \theta < 2\,\pi$.

De este modo, a cada qubit le queda asociado una terna $(1,\ \theta,\ \varphi)$ que representa un punto, en coordenadas esféricas, de la esfera centrada en el origen y de radio uno.
Dicho punto expresado en coordenadas cartesianas es

$$\left(\text{sen}(\varphi)\ \cos(\theta)\,,\ \text{sen}(\varphi)\ \text{sen}(\theta)\,,\ \cos(\varphi)\right).$$

En consecuencia, la función

$$f\,(q) = \left(\text{sen}(\varphi)\ \cos(\theta)\,,\ \text{sen}(\varphi)\ \text{sen}(\theta)\,,\ \cos(\varphi)\right)$$

Establece la correspondencia entre el conjunto de los qubits y la Esfera de Bloch.
Por otro lado, la función *f* es suryectiva, pues:
Dado un punto $P = \left(\text{sen}(\varphi)\ \cos(\theta)\,,\ \text{sen}(\varphi)\ \text{sen}(\theta)\,,\ \cos(\varphi)\right)$ en la Esfera de Bloch, si se define

$$q = \begin{pmatrix} \cos\left(\frac{\varphi}{2}\right) \\ \text{sen}\left(\frac{\varphi}{2}\right)\, e^{i\,\theta} \end{pmatrix}$$

Entonces se prueba fácilmente que f(q) = P ■

Observación 2.2

Cabe preguntarse si la función f de la Proposición 2.1 es inyectiva. La respuesta es que no lo es. Efectivamente, dado un qubit $q = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$, si lo multiplicamos por un factor arbitrario $e^{i\,\delta}$ con $\delta \in \Re$, se puede demostrar fácilmente que f(q) = f(q $e^{i\,\delta}$). Pero este hecho no tiene consecuencias a la hora de "observar o medir" el estado de un qubit, como se verá en la Observación 2.3, al final de esta sección.

¿Qué se puede visualizar en la esfera de Bloch?

Como se expuso anteriormente, la Esfera de Bloch permite visualizar físicamente el estado de un qubit, es decir, la dirección y sentido del eje de rotación del electrón.
En efecto, dado un qubit **q**, el punto **f(q)** de la esfera representa el electrón y el vector $\vec{\mathbf{f}}\mathbf{(q)}$,con origen en **f(q)**, indica la dirección y sentido de su eje de rotación. Cabe destacar que, en la Esfera de Bloch, se suele identificar **f(q)** con **q**.

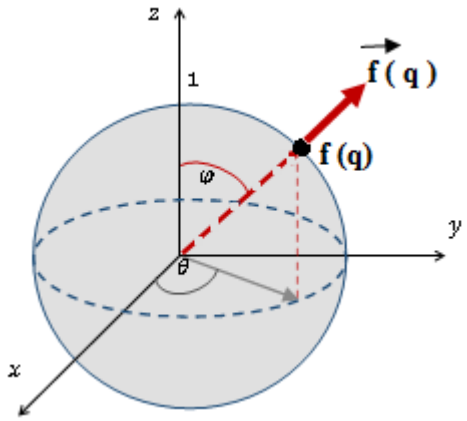


Figura 2.6: Visualización de estado de un qubit en la Esfera de Bloch

En el siguiente ejemplo se mostrará concretamente cómo representar los qubits en la Esfera de Bloch.

Ejemplo 2.1

Representar en la Esfera de Bloch los siguientes qubits:

a) $q = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ b) $q = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ c) $q = \begin{pmatrix} 1/\sqrt{2} \\ i/\sqrt{2} \end{pmatrix}$

Solución

a) En este caso, $\alpha = 1$ y $\beta = 0$.

Primeramente se expresarán α y β en forma exponencial.

Forma exponencial de α

$$\begin{cases} a = Re(\alpha) = 1 \\ b = Im(\alpha) = 0 \end{cases} \Rightarrow \begin{cases} \rho_1 = \sqrt{a^2 + b^2} = 1 \\ \theta_1 = arctg\left(\frac{b}{a}\right) = 0 \end{cases}$$

Forma exponencial de β

$$\begin{cases} a = Re(\beta) = 0 \\ b = Im(\beta) = 0 \end{cases} \Rightarrow \begin{cases} \rho_2 = \sqrt{a^2 + b^2} = 0 \\ \theta_2 = arctg\left(\frac{b}{a}\right) = arctg\left(\frac{0}{0}\right) \end{cases}$$

Según la Observación 2.1, se considera $\theta_2 = 0$.

A continuación se calculan los ángulos φ y θ :

$$\varphi = 2\ \text{arctg}\left(\frac{\rho_2}{\rho_1}\right) = 2\ \text{arctg}\left(\frac{0}{1}\right) = 2\ \text{arctg}(0) = 0$$

$$\theta = \theta_2 - \theta_1 = 0 - 0 = 0$$

Por lo tanto, si $q = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ entonces

$$f(q) = \left(\text{sen}(\varphi)\cos(\theta), \text{sen}(\varphi)\,\text{sen}(\theta), \cos(\varphi)\right) = (0, 0, 1).$$

Es decir que el qubit $q = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = |0>$ se representa en la Esfera de Bloch mediante el punto (0, 0, 1). Además, el vector $\vec{f}(q)$ indica la dirección y sentido del eje de rotación del electrón, el cual está **spín arriba** y en consecuencia, se supone que está en estado **0**. Por esta razón se considera que, cuando **q = | 0 >** , hay almacenado un **0**.

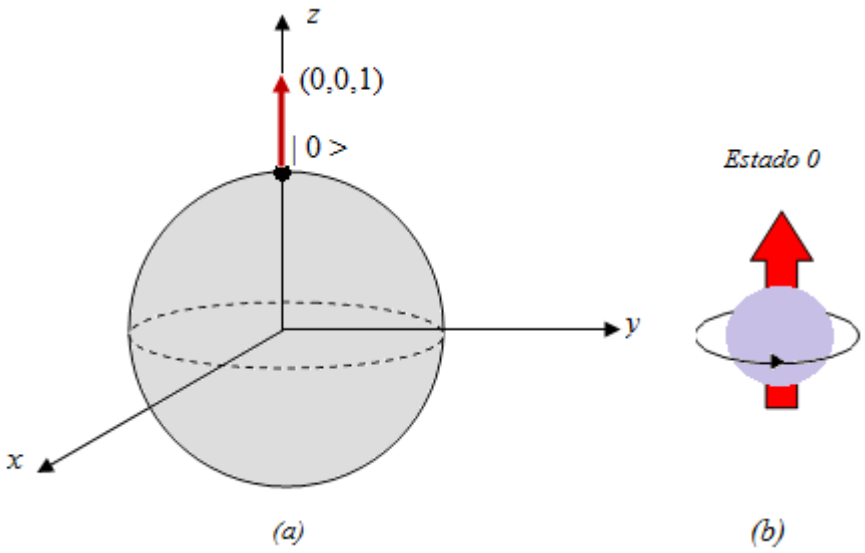


Figura 2.7: (a) Representación del qubit q = |0> en la Esfera de Bloch.
(b) El vector f(q) indica que el electrón está spin arriba o en estado 0.

b) En este caso, $\alpha = 0$ y $\beta = 1$.

Nuevamente se expresan α y β en forma exponencial.

Forma exponencial de α

$$\begin{cases} a = \text{Re}(\alpha) = 0 \\ b = \text{Im}(\alpha) = 0 \end{cases} \Rightarrow \begin{cases} \rho_1 = \sqrt{a^2 + b^2} = 0 \\ \theta_1 = \text{arctg}\left(\frac{b}{a}\right) = \text{arctg}\left(\frac{0}{0}\right) \end{cases}$$

Por lo tanto, $\theta_1 = 0$

Forma exponencial de β

$$\begin{cases} a = \text{Re}(\beta) = 1 \\ b = \text{Im}(\beta) = 0 \end{cases} \Rightarrow \begin{cases} \rho_2 = \sqrt{a^2 + b^2} = 1 \\ \theta_2 = \text{arctg}\left(\frac{b}{a}\right) = \text{arctg}(0) = 0 \end{cases}$$

Ahora se calculan los ángulos φ y θ :

$$\varphi = 2\ \text{arctg}\left(\frac{\rho_2}{\rho_1}\right) = 2\ \text{arctg}\left(\frac{1}{0}\right) = 2\ \frac{\pi}{2} = \pi$$

$$\theta = \theta_2 - \theta_1 = 0 - 0 = 0$$

Por lo tanto, si $q = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ entonces

$$f(q) = \left(\text{sen}(\varphi)\cos(\theta), \text{sen}(\varphi)\,\text{sen}(\theta), \cos(\varphi)\right) = (0, 0, -1).$$

Es decir que el qubit $q = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = |1>$ se representa en la Esfera de Bloch mediante el punto (0, 0, - 1). Además, el vector $\vec{f}(q)$ indica la dirección y sentido del eje de rotación del electrón, el cual se encuentra **spín abajo** y en consecuencia, se supone que está en estado **1**.
Por esta razón se considera que, cuando **q = | 1 >** , hay almacenado un **1**.

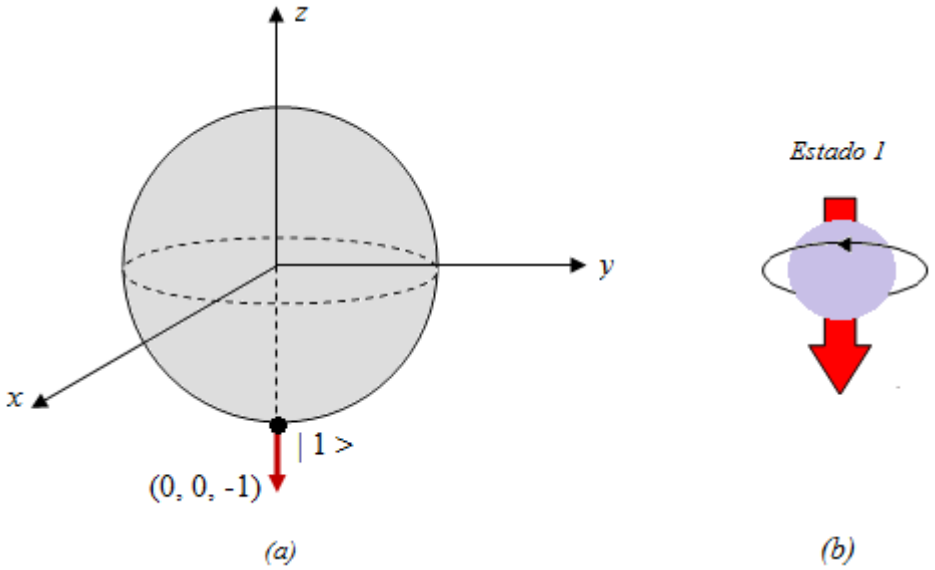


Figura 2.8: (a) Representación del qubit q = |1> en la Esfera de Bloch.
(b) El vector f(q) indica que el electrón está spin arriba o en estado 1.

c) En este caso, $\alpha = \frac{1}{\sqrt{2}}$ y $\beta = \frac{i}{\sqrt{2}}$

Forma exponencial de α

$$\begin{cases} a = \text{Re}(\alpha) = 1/\sqrt{2} \\ b = \text{Im}(\alpha) = 0 \end{cases} \Rightarrow \begin{cases} \rho_1 = \sqrt{a^2 + b^2} = 1/\sqrt{2} \\ \theta_1 = \text{arctg}\left(\frac{b}{a}\right) = \text{arctg}(0) = 0 \end{cases}$$

Forma exponencial de β

De manera similar, resulta

$$\begin{cases} \rho_2 = \frac{1}{\sqrt{2}} \\ \theta_2 = \frac{\pi}{2} \end{cases}$$

Los ángulos φ y θ son:

$$\varphi = 2 \text{ arctg} \left(\frac{\rho_2}{\rho_1} \right) = 2 \text{ arctg } (1) = 2 \frac{\pi}{4} = \frac{\pi}{2}$$

$$\theta = \theta_2 - \theta_1 = \frac{\pi}{2}$$

Por lo tanto, si $q = \begin{pmatrix} 1/\sqrt{2} \\ i/\sqrt{2} \end{pmatrix}$ entonces

$$f(q) = \left(\text{sen}(\varphi) \cos(\theta) , \text{sen}(\varphi) \text{sen}(\theta) , \cos(\varphi) \right) = (0, 1, 0).$$

Es decir que el qubit $q = \begin{pmatrix} 1/\sqrt{2} \\ i/\sqrt{2} \end{pmatrix} = \frac{1}{\sqrt{2}} \mid 0 > + \frac{i}{\sqrt{2}} \mid 1 >$ se representa en la Esfera de Bloch mediante el punto (0, 1, 0). El electrón está girando en la dirección y sentido que indica el vector $\vec{f}(q)$. Como α y β son no nulos, el qubit está en estado de superposición, lo cual indica que contiene ambos valores: **0** y **1**.

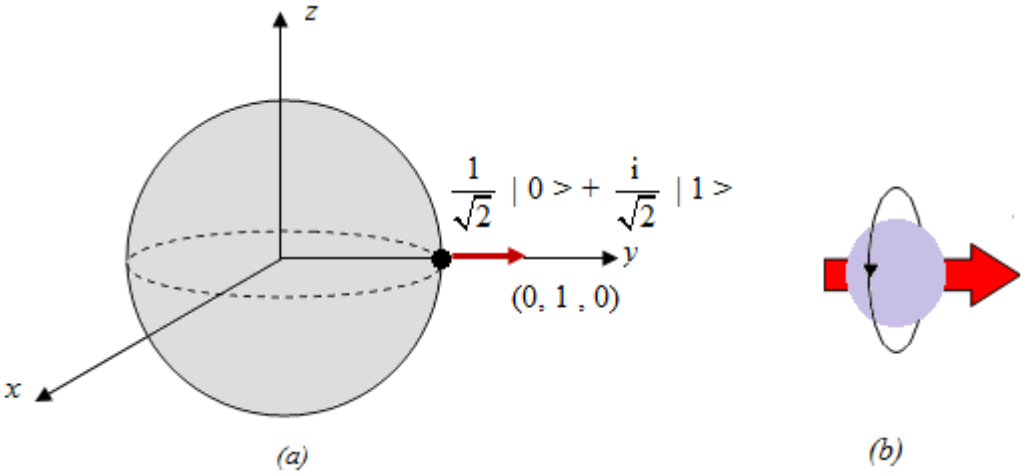


Figura 2.9: (a) Representación del qubit q en la Esfera de Bloch.
(b) El vector f(q) determina el eje de rotación del electrón .

Simulación del estado de un qubit

La siguiente simulación, creada con el software Wolfram Mathematica 9.0, muestra el giro del electrón, según el estado del qubit $q = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$ donde $\begin{cases} \alpha = \rho_1 \, e^{i \theta_1} \\ \beta = \rho_2 \, e^{i \theta_2} \end{cases}$.

El electrón se representó mediante un cubo y su eje de rotación se calculó en base a la expresión de la función f(q), definida en la Proposición 2.1

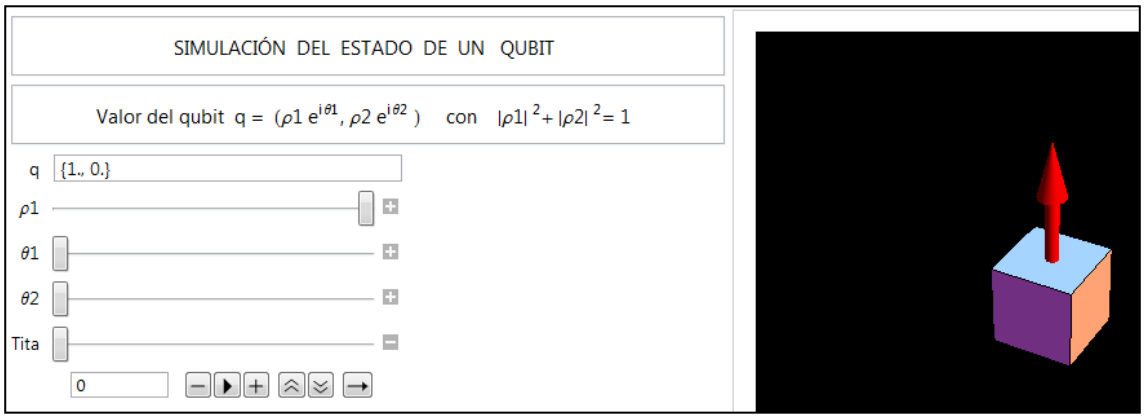


Figura 2.10: Simulación del estado de un qubit

El código fuente correspondiente a dicha simulación se muestra a continuación:

```
Manipulate[ ρ2 = √(1 - ρ1^2) ;

 fi = If[ρ1 ≠ 0, 2 ArcTan[ρ2/ρ1] , 2 Pi / 2];

 tita = θ2 - θ1;
 q = {ρ1 (Cos[θ1] + i Sin[θ1]), ρ2 (Cos[θ2] + i Sin[θ2])};
 q1 = { Sin[fi] Cos[tita], Sin[ fi] Sin[tita], Cos[fi] };
 Cubo = Graphics3D[Rotate[Cuboid[{-0.5, -0.5, -0.5}], Tita Degree, q1], ImageSize → Small];
 Eje = Graphics3D[{Red, Arrowheads[0.1], Arrow[Tube[{{0, 0, 0}, 2 q1}, 0.08]] }];
 Show[Eje , Cubo, Boxed → False, Background → Black, PlotRange → {{-2, 2}, {-2, 2}, {-2, 2}}],
 Panel["                              SIMULACIÓN DEL ESTADO DE UN QUBIT                    "],
 Panel[ "          Valor del qubit q = (ρ1 e^iθ1, ρ2 e^iθ2)      con     |ρ1|^2 + |ρ2|^2 = 1        "],
 {q},
 {ρ1, 0, 1}, {θ1, 0, 2 Pi}, {θ2, 0, 2 Pi}, {Tita, 0, 360, AnimationRate → 160}, ControlPlacement → Left ]
```

Figura 2.11: Código fuente de la simulación del estado de un qubit

La Esfera de Bloch es también utilizada para visualizar el cambio de estado de un qubit al atravesar una compuerta cuántica. Esto se mostrará en el Capítulo 3.

¿Cómo medir el estado de un qubit ?

Según el Principio de Superposición de la Mecánica Cuántica, mientras una partícula subatómica no interactúa con su entorno, ésta puede estar en varios estados al mismo tiempo. Es decir, se comporta como si pudiera hacer varias cosas simultáneamente. En particular, un qubit $q = \alpha\ |0> + \beta\ |1>$ puede estar en ambos estados: 0 y 1.

Pero curiosamente, en el instante en que es sometido a una medición, el qubit "colapsa" a uno de estos dos estados: 0 o bien 1. Esto significa que el estado de superposción desaparece y ello se debe a que, en el momento en que el electrón es observado, sólo adopta una de las dos posiciones de giro: spin arriba (estado 0) o spin abajo (estado 1).

Pero, ¿cómo saber a cuál de estos dos estados colapsará el qubit inmediatamente antes de efectuar la medición? Sólo se puede saber en términos de probabilidades y, para calcularlas, se necesita aplicar el Tercer Postulado de la Mecánica Cuántica, que excede el objetivo de este trabajo. Por esta razón los resultados del cálculo de estas probabilidades serán dados en términos de la siguiente definición:

Definición 2.3: Medida de un qubit

Dado un sistema formado por un qubit, cuyo estado se describe mediante el vector

$$\mathbf{q} = \alpha \,|\, \mathbf{0} > + \beta \;\; |\, \mathbf{1} >$$

El número $|\alpha|^2$ *indica la probabilidad de encontrar el qubit del sistema en estado 0 .*

Mientras que $|\beta|^2$ *da la probabilidad de encontrar el qubit del sistema en estado 1.*

Observación 2.3

Como se mencionó en la Observación 2.2, a los qubits $q = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$ y $q\ e^{i\delta} = \begin{pmatrix} \alpha\, e^{i\delta} \\ \beta\, e^{i\delta} \end{pmatrix}$ con $\delta \in \Re$ les corresponde el mismo punto en la Esfera de Bloch. Sin embargo, este hecho no tiene consecuencias observables, ya que las únicas cantidades "medibles" del qubit q son las probabilidades $|\alpha|^2$ y $|\beta|^2$, mientras que las del qubit $q\ e^{i\delta}$ son

$$\left|\alpha\, e^{i\delta}\right|^2 = \overline{\left(\alpha\, e^{i\delta}\right)}\left(\alpha\, e^{i\delta}\right) = \bar{\alpha}\,\overline{e^{i\delta}}\;\alpha\, e^{i\delta} = \bar{\alpha}\, e^{-i\delta}\alpha\, e^{i\delta} = \bar{\alpha}\,\alpha\, e^{-i\delta}e^{i\delta} = \bar{\alpha}\,\alpha = |\alpha|^2$$

$$\left|\beta\, e^{i\delta}\right|^2 = \overline{\left(\beta\, e^{i\delta}\right)}\left(\beta\, e^{i\delta}\right) = \bar{\beta}\,\overline{e^{i\delta}}\;\beta\, e^{i\delta} = \bar{\beta}\, e^{-i\delta}\beta\, e^{i\delta} = \bar{\beta}\,\beta\, e^{-i\delta}e^{i\delta} = \bar{\beta}\,\beta = |\beta|^2$$

Por lo tanto, al multiplicar el qubit q por un factor $e^{i\delta}$, no altera los valores $|\alpha|^2$ y $|\beta|^2$**.** En consecuencia, no modifica el valor de la medición de q. ■

Cabe preguntarse qué pasaría si las computadoras cuánticas utilizaran, para procesar los datos, más de un qubit. En estos casos los qubits comienzan a interactuar entre sí para generar un nuevo sistema físico. En la próxima sección se mostrará cómo describir su estado. Se verá que el producto tensorial desempeñará un rol importante, ya que permitirá modelar matemáticamente este tipo de sistemas.

2.2 Sistemas formados por varios qubits

Para describir el estado de un sistema formado por varios qubits, se recurrirá al Cuarto Postulado de la Mecánica Cuántica:

Cuarto Postulado de la Mecánica Cuántica:

El espacio de estados de un sistema compuesto por dos o más sistemas físicos es el producto tensorial de los espacios de estado de cada uno de sus componentes.
Más específicamente, si los sistemas físicos están numerados de 1 a n y si el estado de cada sistema es q_i , con i = 1,...,n entonces el estado del sistema compuesto será

$$q_1 \otimes q_2 \otimes .... q_n$$

Por razones didácticas se comenzará describiendo los sistemas formados por 2 qubits, luego por 3 qubits, para finalmente generalizarlos a n qubits.

Sistemas formados por dos qubits

Según el Cuarto Postulado de la Mecánica Cuántica, el estado de un sistema formado por dos qubits q_1 y q_2 queda determinado por el producto tensorial $q_1 \otimes q_2$.
A fin de simplificar la expresión de los productos tensoriales, se introducirá la siguiente notación:

Notación 2.1:
Si $x, y \in \{ 0 , 1 \}$ entonces se escribe

$$| x > \otimes | y > = | x\ y > \quad \text{o bien} \quad | x > \otimes | y > = | x , y >$$

Por lo tanto, si

$$q_1 = \alpha_1 | 0 > + \beta_1 | 1 > \quad y \quad q_2 = \alpha_2 | 0 > + \beta_2 | 1 >$$

Entonces, al aplicar las propiedades del producto tensorial resulta

$$q_1 \otimes q_2 = (\alpha_1 | 0 > + \beta_1 | 1 >) \otimes (\alpha_2 | 0 > + \beta_2 | 1 >) =$$

$$= \alpha_1 \alpha_2 | 0\ 0 > + \alpha_1 \beta_2 | 0\ 1 > + \beta_1 \alpha_2 | 1\ 0 > + \beta_1 \beta_2 | 1\ 1 >$$

Al calcular estos productos tensoriales se obtiene

$$| 0\ 0 > = | 0 > \otimes | 0 > = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \begin{pmatrix} 1 \\ 0 \end{pmatrix} \\ 0 \begin{pmatrix} 1 \\ 0 \end{pmatrix} \end{pmatrix} = \begin{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} \\ \begin{pmatrix} 0 \\ 0 \end{pmatrix} \end{pmatrix} \approx \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$| 0\,1 > = | 0 > \otimes | 1 > = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1\begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ 0\begin{pmatrix} 0 \\ 1 \end{pmatrix} \end{pmatrix} = \begin{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ \begin{pmatrix} 0 \\ 0 \end{pmatrix} \end{pmatrix} \approx \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

$$| 1\,0 > = | 1 > \otimes | 0 > = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0\begin{pmatrix} 1 \\ 0 \end{pmatrix} \\ 1\begin{pmatrix} 1 \\ 0 \end{pmatrix} \end{pmatrix} = \begin{pmatrix} \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\ \begin{pmatrix} 1 \\ 0 \end{pmatrix} \end{pmatrix} \approx \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

$$| 1\,1 > = | 1 > \otimes | 1 > = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0\begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ 1\begin{pmatrix} 0 \\ 1 \end{pmatrix} \end{pmatrix} = \begin{pmatrix} \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\ \begin{pmatrix} 0 \\ 1 \end{pmatrix} \end{pmatrix} \approx \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

En consecuencia,

$$q_1 \otimes q_2 = \begin{pmatrix} \alpha_1\ \alpha_2 \\ \alpha_1\ \beta_2 \\ \beta_1\ \alpha_2 \\ \beta_1\ \beta_2 \end{pmatrix}$$

Como se puede comprobar, el producto tensorial de dos qubits genera un vector de 2^2 componentes. Por otro lado, según el Primer Postulado de la Mecánica Cuántica, el vector de estado del sistema debe ser de módulo 1. Esto da lugar a definir un 2-qubit de la siguiente manera:

Definición 2.4: 2-qubit

Un 2- qubit es un vector $\mathbf{q} \in \mathbb{C}^4$ *de módulo 1. Es decir,*

$$\mathbf{q} = \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{pmatrix}$$

Donde $a_0, a_1, a_2, a_3 \in \mathbb{C}$ *y verifican que* $\sum_{i=0}^{3} | a_i |^2 = 1$

Teniendo en cuenta que $\{ | 00 >, | 01 >, | 10 >, | 11 > \}$ es una base para el espacio de los 2- qubits, la Definición 2.4 se puede reformular de la siguiente manera:

Definición 2.5:

Un 2- qubit es un vector **q** *de la forma*

$$\mathbf{q} = a_0 \,|\, \mathbf{0\ 0} > + a_1 \,|\, \mathbf{0\ 1} > + a_2 \,|\, \mathbf{1\ 0} > + a_3 \,|\, \mathbf{1\ 1} >$$

Donde

$$a_0, a_1, a_2, a_3 \in \mathbb{C} \quad y \quad \sum_{i=0}^{3} |\, a_i \,|^2 = 1$$

En este caso, el conjunto $\{\, |00>, |01>, |10>, |11> \,\}$ *constituye una base para el espacio de los 2- qubits*

De este modo, un 2-qubit es un vector que describe el estado de un sistema formado por dos qubits. Nuevamente, del Tercer Postulado de la Mecánica Cuántica se deduce la siguiente definición:

Definición 2.6: Medida de un 2- qubit

Dado un sistema formado por dos qubits, cuyo estado se describe mediante el 2-qubit

$$\mathbf{q} = a_0 \,|\, \mathbf{0\ 0} > + a_1 \,|\, \mathbf{0\ 1} > + a_2 \,|\, \mathbf{1\ 0} > + a_3 \,|\, \mathbf{1\ 1} >$$

Entonces

$|\, a_0 \,|^2$ *indica la probabilidad de encontrar el primer y segundo qubit del sistema en estado 0.*

$|\, a_1 \,|^2$ *indica la probabilidad de encontrar el primer qubit del sistema en estado 0 y el segundo qubit del sistema en estado 1.*

$|\, a_2 \,|^2$ *indica la probabilidad de encontrar el primer qubit del sistema en estado 1 y el segundo qubit del sistema en estado 0.*

$|\, a_3 \,|^2$ *indica la probabilidad de encontrar el primer y segundo qubit del sistema en estado 1.*

Sistemas formados por tres qubits

El estado de un sistema formado por tres qubits q_1 , q_2 y q_3 queda determinado, según el Cuarto Postulado de la Física Cuántica, por el producto tensorial $q_1 \otimes q_2 \otimes q_3$.
Nuevamente, a fin de simplificar la expresión de los productos tensoriales, se introduce la siguiente notación:

Notación 2.2:

Si $x, y, z \in \{\, 0\ ,\ 1 \,\}$ entonces se escribe

$|x> \otimes |y> \otimes |z> = |x\ y\ z>$ o bien $|x> \otimes |y> \otimes |z> = |x\ y> |z>$

Por lo tanto, si

$$q_1 = \alpha_1 |0> + \beta_1 |1> \ , \ q_2 = \alpha_2 |0> + \beta_2 |1> \ \text{ y } \ q_3 = \alpha_3 |0> + \beta_3 |1>$$

Entonces, al aplicar las propiedades del producto tensorial se obtiene

$$q_1 \otimes q_2 \otimes q_3 = (\alpha_1 |0> + \beta_1 |1>) \otimes (\alpha_2 |0> + \beta_2 |1>) \otimes (\alpha_3 |0> + \beta_3 |1>)$$

$$= \alpha_1 \alpha_2 \alpha_3 |0\,0\,0> + \alpha_1 \alpha_2 \beta_3 |0\,0\,1> + \alpha_1 \beta_2 \alpha_3 |0\,1\,0> + \alpha_1 \beta_2 \beta_3 |0\,1\,1> +$$

$$+ \beta_1 \beta_2 \alpha_3 |1\,0\,0> + \beta_1 \alpha_2 \beta_3 |1\,0\,1> + \beta_3 \beta_2 \alpha_3 |1\,1\,0> + \beta_1 \beta_2 \beta_3 |1\,1\,1>$$

Al calcular estos productos tensoriales resulta

$$|0\,0\,0> = |0> \otimes |0> \otimes |0> = \begin{pmatrix}1\\0\end{pmatrix} \otimes \begin{pmatrix}1\\0\end{pmatrix} \otimes \begin{pmatrix}1\\0\end{pmatrix} \approx \begin{pmatrix}1\\0\\0\\0\end{pmatrix} \otimes \begin{pmatrix}1\\0\end{pmatrix} = \begin{pmatrix} 1\begin{pmatrix}1\\0\end{pmatrix} \\ 0\begin{pmatrix}1\\0\end{pmatrix} \\ 0\begin{pmatrix}1\\0\end{pmatrix} \\ 0\begin{pmatrix}1\\0\end{pmatrix} \end{pmatrix} =$$

$$= \begin{pmatrix} \begin{pmatrix}1\\0\end{pmatrix} \\ \begin{pmatrix}0\\0\end{pmatrix} \\ \begin{pmatrix}0\\0\end{pmatrix} \\ \begin{pmatrix}0\\0\end{pmatrix} \end{pmatrix} \approx \begin{pmatrix}1\\0\\0\\0\\0\\0\\0\\0\end{pmatrix}$$

Y así para los demás casos. En consecuencia,

$$q_1 \otimes q_2 \otimes q_3 = \begin{pmatrix} \alpha_1\ \alpha_2\ \alpha_3 \\ \alpha_1\ \alpha_2\ \beta_3 \\ \alpha_1\ \beta_2\ \alpha_3 \\ \alpha_1\ \beta_2\ \beta_3 \\ \beta_1\ \beta_2\ \alpha_3 \\ \beta_1\ \alpha_2\ \beta_3 \\ \beta_3\ \beta_2\ \alpha_3 \\ \beta_3\ \beta_2\ \beta_3 \end{pmatrix}$$

De este modo, el producto tensorial de 3 qubits genera un vector de 2^3 componentes. Además, por el Primer Postulado de la Física Cuántica, el vector de estado del sistema debe tener módulo 1. Esto da lugar a definir un 3-qubit de la siguiente manera:

Definición 2.7: 3-qubit

Un 3- qubit es un vector $\mathbf{q} \in \mathbb{C}^8$ *de módulo 1. Es decir,*

$$\mathbf{q} = \left(a_0 , a_1, a_2, a_3, a_4, a_5, a_6, a_7 \right)^T$$

Donde $a_i \in \mathbb{C} \ \forall\, i = 1,...,7$ *y verifican que* $\sum_{i=0}^{7} | a_i |^2 = 1$

Teniendo en cuenta que

$$\{ | 000> , | 001 > , | 010 > , | 011 > , | 100> , | 101> , | 110> , |111> \}$$

Es una base para el espacio de los 3- qubits, la Definición 2.7 se reescribe de la siguiente manera:

Definición 2.8:

Un 3- qubit es un vector **q** *de la forma*

$$\mathbf{q} = a_0 \,|\mathbf{0\,0\,0}> + a_1 \,|\mathbf{0\,0\,1}> + a_2 \,|\mathbf{0\,1\,0}> + a_3 \,|\mathbf{0\,1\,1}> + a_4 \,|\mathbf{1\,0\,0}> + a_5 \,|\mathbf{1\,0\,1}> + a_6 \,|\mathbf{1\,1\,0}> + a_7 \,|\mathbf{1\,1\,1}>$$

Donde

$$a_i \in \mathbb{C} \ \forall\, i = 1,...,7 \quad y \quad \sum_{i=0}^{7} | a_i |^2 = 1$$

En este caso, el conjunto

$$\{ | 000 > , | 001 > , | 010 > , | 011 > , | 100 > , | 101 > , | 110 > , |111 > \}$$

Constituye una base para el espacio de los 3- qubits

De este modo, un 3-qubit es un vector que describe el estado de un sistema formado por tres qubits. La medida de un 3-qubit se define de manera similar a las Definiciones 2.3 y 2.6.
A continuación se definirán los n- qubits, los cuales permitirán describir el estado de un sistema formado por n qubits.

Sistemas formados por n qubits

El estado de un sistema formado por n qubits se describe mediante un n-qubit. Su definición es la siguiente:

Definición 2.9: n-qubit

Dado un número natural n, un n- qubit es un vector $\mathbf{q} \in \mathbb{C}^{2^n}$ *de módulo 1. Es decir,*

$$\mathbf{q} = \left(a_0 , a_1, \ldots , a_{2^n-1} \right)^T$$

Donde $a_i \in \mathbb{C} \ \forall\, i = 0, \ldots , 2^n - 1$ *y verifican que* $\sum_{i=0}^{2^n-1} | a_i |^2 = 1$

En este caso, una base del espacio de los n- qubits es

$$\{ | x_{n-1}\, x_{n-2} \ldots x_1\, x_0 > \text{ con } x_i = 0 \text{ ó } 1 ,\ i = 0, 1 ,..., n - 1 \}$$

Donde

$$| x_{n-1}\, x_{n-2} \ldots x_1\, x_0 > = | x_{n-1} > \otimes | x_{n-2} > \otimes \ldots \otimes | x_1 > \otimes | x_0 >$$

Si se quisiera representar un n-qubit como combinación lineal de los elementos de esta base (de una manera similar a la planteada en las Definiciones 2.5 y 2.8) entonces su expresión, con esta notación, se tornaría complicada.
Por esta razón se adoptará la siguiente convención, muy utilizada en Computación Cuántica:

Notación 2.3:

Si $x_i = 0$ ó 1 con $i = 0, 1 ,..., n - 1$ *entonces*

$$| x_{n-1}\, x_{n-2} \ldots x_1\, x_0 > \equiv | x >$$

Donde x *es la representación en base 10 del número binario* $x_{n-1}\, x_{n-2} \ldots x_1\, x_0$.
Es decir,

$$x = 2^0\, x_0 + 2^1\, x_1 + \ldots + 2^{n-2}\, x_{n-2} + 2^{n-1}\, x_{n-1}$$

El siguiente ejemplo ilustrará cómo aplicar esta notación.

Ejemplo 2.2

Dado el 2-qubit $q = a_0 | 0\ 0 > + a_1 | 0\ 1 > + a_2 | 1\ 0 > + a_3 | 1\ 1 >$, expresarlo según la Notación 2.3

Solución

El siguiente cuadro ilustra cómo representar cada elemento de la base del espacio de los 2-qubits:

Número binario	***Representación en base 10***
0 0	$0\,(2^0) + 0\,(2^1) = 0$
0 1	$1\,(2^0) + 0\,(2^1) = 1$
1 0	$0\,(2^0) + 1\,(2^1) = 2$
1 1	$1\,(2^0) + 1\,(2^1) = 3$

Por lo tanto

$$| 0\ 0 > \equiv | 0 >$$
$$| 0\ 1 > \equiv | 1 >$$
$$| 1\ 0 > \equiv | 2 >$$
$$| 1\ 1 > \equiv | 3 >$$

En consecuencia, el 2-qubit q se escribe

$$q = \sum_{i=0}^{3} a_i \,| i > \quad \blacksquare$$

Con esta nueva notación para los elementos de la base del espacio de los n-qubits, se puede extender las Definiciones 2.5 y 2.8 del siguiente modo:

Definición 2.10:

Un n- qubit es un vector **q** *de la forma*

$$\mathbf{q} = \sum_{i=0}^{2^n-1} a_i \,| i >$$

Donde

$$a_i \in \mathbb{C} \quad \forall\, i = 0,..., 2^n - 1 \quad y \quad \sum_{i=0}^{2^n-1} |\, a_i \,|^2 = 1$$

En este caso, el conjunto

$$\left\{ | 0 > ,\ | 1 > , | 2 > , \ldots . ,\ | 2^n - 1 > \right\}$$

Constituye una base para el espacio de los n- qubits

En el capítulo introductorio se hizo una somera explicación física acerca del entrelazamiento cuántico, un fenómeno fascinante que no deja de sorprender a la ciencia. En la siguiente sección se verá, mediante un ejemplo, cómo se justifica matemáticamente este hecho.

2.3 El entrelazamiento cuántico visto desde el punto de vista matemático

Como se mencionó en el capítulo introductorio, las partículas subatómicas cuentan con una propiedad llamada **entrelazamiento o enredo cuántico** (**entanglement** en inglés).
Este fenómeno funciona del siguiente modo:
Cuando dos o más partículas subatómicas interactúan, éstas pueden acabar entrelazadas. Esto significa que su posición u otras propiedades quedan estrechamente vinculadas mediante un proceso desconocido, hasta el día de hoy, por la ciencia moderna.
Gracias a esta particularidad, lo que le ocurre a una partícula puede afectar a la otra, aunque disten entre sí millones de años luz y no haya nada de por medio entre ellas.
Según Einstein, "***Éste es un fenómeno espeluznante a distancia….***"
Como el estado de las partículas cuánticas se describe mediante los qubits, el entrelazamiento cuántico hace que cualquier cambio en el estado de un qubit, provoque un cambio instantáneo en el otro qubit entrelazado. Pero ¿cómo se describe matemáticamente este fenómeno? Para responder a esta pregunta, se dará primeramente su correspondiente definición.

Definición 2.11: Qubits entrelazados

Dado un sistema cuántico formado por n qubits $\mathbf{q_1, q_2, \ldots, q_n}$ *, se dice que éstos están entrelazados si el n-qubit* $\mathbf{q}$ *,que describe el estado del sistema, no puede ser expresado como producto tensorial de los qubits que componen dicho sistema.*
Más específicamente, $\mathbf{q_1, q_2, \ldots, q_n}$ *están entrelazados si*

$$\mathbf{q \neq q_1 \otimes q_2 \otimes \ldots \otimes q_n}$$

Por el contrario, si

$$\mathbf{q = q_1 \otimes q_2 \otimes \ldots \otimes q_n}$$

Se dirá que los qubits $\mathbf{q_1, q_2, \ldots, q_n}$ no están entrelazados.

Ejemplo 2.3

Probar que del 2-qubit $q = \frac{1}{\sqrt{2}} |0\,1> + \frac{1}{\sqrt{2}} |1\,0>$ se deduce que los dos qubits q_1 y q_2 que componen el sistema están entrelazados.

Solución

Supongamos por el absurdo que q_1 y q_2 no están entrelazados, es decir,

$$q = q_1 \otimes q_2$$

Donde

$$q_1 = a\,|\,0\,> + b\,|\,1\,>$$
$$q_2 = c\,|\,0\,> + d\,|\,1\,>$$

Por lo tanto

$$q = q_1 \otimes q_2 \Rightarrow$$

$$\frac{1}{\sqrt{2}}\,|\,0\,1\,> + \frac{1}{\sqrt{2}}\,|\,1\,0\,> = a\,c\,|\,0\,0\,> + a\,d\,|\,0\,1\,> + b\,c\,|\,1\,0\,> + b\,d\,|\,1\,1\,>$$

De aquí resulta el sistema de ecuaciones

$$\begin{cases} a\,c = 0 \\ a\,d = \dfrac{1}{\sqrt{2}} \\ b\,c = \dfrac{1}{\sqrt{2}} \\ b\,d = 0 \end{cases}$$

De la primera ecuación se tiene que a = 0 ó c = 0.

Si a = 0, al reemplazar en la segunda ecuación se obtiene $0 = \frac{1}{\sqrt{2}}$.

Similarmente, si c = 0 , al reemplazar en la tercera ecuación resulta $0 = \frac{1}{\sqrt{2}}$.

En ambos casos se llega a un absurdo, que provino de suponer que los qubits no estaban entrelazados. ■

Ejemplo 2.4

Probar que el 2-qubit **q =** $\frac{1}{2}$ **(| 0 0 > - | 0 1 > + | 1 0 > - | 1 1 >)** indica que los dos qubits q_1 y q_2 que componen el sistema no están entrelazados.

Solución

Se tratará de hallar q_1 y q_2 de modo tal que

$$q = q_1 \otimes q_2$$

Donde

$$q_1 = a\,|\,0\,> + b\,|\,1\,>$$
$$q_2 = c\,|\,0\,> + d\,|\,1\,>$$

Por lo tanto

$$q = q_1 \otimes q_2 \Rightarrow$$

$$\frac{1}{2}\left(|\ 00\ > - |\ 01\ > + |\ 10\ > - |\ 11\ > \right) = a\ c\ |\ 00\ > + a\ d\ |\ 01\ > + b\ c\ |\ 10\ > + b\ d\ |\ 11\ >$$

De aquí resulta el sistema de ecuaciones

$$\begin{cases} a\ c = \frac{1}{2} \\ a\ d = \frac{1}{\sqrt{2}} \\ b\ c = \frac{1}{\sqrt{2}} \\ b\ d = -\frac{1}{2} \end{cases}$$

Que tiene la solución $a = b = c = \frac{1}{\sqrt{2}}$, $d = -\frac{1}{\sqrt{2}}$.

Por lo tanto, si

$$q_1 = \frac{1}{\sqrt{2}}\ |\ 0\ > + \frac{1}{\sqrt{2}}\ |\ 1\ >$$

$$q_2 = \frac{1}{\sqrt{2}}\ |\ 0\ > - \frac{1}{\sqrt{2}}\ |\ 1\ >$$

Entonces se verifica que

$$q = q_1 \otimes q_2 \quad \blacksquare$$

Se sabe que, al medir el estado de un qubit del sistema, únicamente es posible obtener dos resultados: estado 0 o bien estado 1. Además, antes de efectuar la medición, sólo se puede predecir en términos de probabilidades en qué estado se hallará dicho qubit.
Por esta razón, para continuar desarrollando este tema, se utilizará como recurso el cálculo de probabilidades:

Sea X_i la variable aleatoria que resulta de medir el estado del i-ésimo qubit de un sistema compuesto por n qubits. Entonces

$$X_i = \{\ 0\ ,\ 1\ \} \quad i = 1, 2\ , \ldots\ , n$$

Por otro lado, se define

$P\ (\ X_i = x)$: La probabilidad de encontrar el i-ésimo qubit del sistema en estado x
(con x = 0 ó x = 1)

Por razones didácticas, se supondrá que el sistema cuántico está formado por dos qubits. La siguientes Proposiciones serán utilizadas para probar el efecto del entrelazamiento cuántico.

Proposición 2.2

Sea un sistema compuesto por dos qubits, y cuyo estado se describe mediante el 2-qubit

$$\mathbf{q} = a_0 \,|\mathbf{0\,0}\rangle + a_1 \,|\mathbf{0\,1}\rangle + a_2 \,|\mathbf{1\,0}\rangle + a_3 \,|\mathbf{1\,1}\rangle$$

Entonces

1) $P(X_1 = 0) = |a_0|^2 + |a_1|^2$
2) $P(X_1 = 1) = |a_2|^2 + |a_3|^2$
3) $P(X_2 = 0) = |a_0|^2 + |a_2|^2$
4) $P(X_2 = 1) = |a_1|^2 + |a_3|^2$

Demostración

1) Al aplicar las propiedades del cálculo de probabilidades resulta

$P(X_1 = 0) = P(X_1 = 0 \wedge (X_2 = 0 \vee X_2 = 1)) =$

$= P((X_1 = 0 \wedge X_2 = 0) \vee (X_1 = 0 \wedge X_2 = 1)) =$

$= P(X_1 = 0 \wedge X_2 = 0) + P(X_1 = 0 \wedge X_2 = 1)$

De acuerdo a la Definición 2.6 se obtiene

$P(X_1 = 0 \wedge X_2 = 0) = |a_0|^2$, $P(X_1 = 0 \wedge X_2 = 1) = |a_1|^2$

Por lo tanto,

$$P(X_1 = 0) = |a_0|^2 + |a_1|^2$$

Los incisos 2) , 3) y 4) se demuestran de manera similar. ■

Proposición 2.3

Sea un sistema compuesto por dos qubits, y cuyo estado se describe mediante el 2-qubit

$$\mathbf{q} = a_0 \,|\mathbf{0\,0}\rangle + a_1 \,|\mathbf{0\,1}\rangle + a_2 \,|\mathbf{1\,0}\rangle + a_3 \,|\mathbf{1\,1}\rangle$$

Entonces

Si una vez realizada la medición sólo del primer qubit del sistema, se obtiene que éste está en estado 0, el 2-qubit evoluciona al siguiente estado:

$$\mathbf{q} = \frac{\mathbf{a_0}\,|\mathbf{0\,0}\rangle + \mathbf{a_1}\,|\mathbf{0\,1}\rangle}{\sqrt{|\mathbf{a_0}|^2 + |\mathbf{a_1}|^2}}$$

Y si se obtiene que éste está en estado 1, el 2-qubit evoluciona a un nuevo estado dado por

$$\mathbf{q} = \frac{\mathbf{a_2}\,|\mathbf{1\,0}\rangle + \mathbf{a_3}\,|\mathbf{1\,1}\rangle}{\sqrt{|\mathbf{a_2}|^2 + |\mathbf{a_3}|^2}}$$

Expresiones similares se obtienen para los resultados de la medida del segundo qubit del sistema.

Demostración

Resulta de aplicar el Tercer Postulado de la Mecánica Cuántica. La demostración no será desarrollada ya que los detalles de la misma van más allá del objetivo de este trabajo. ■

La proposición anterior se puede extender a un sistema formado por tres qubits, como sigue:
Los incisos 2) , 3) y 4) se demuestran de manera similar. ■

Proposición 2.4

Sea un sistema compuesto por tres qubits, y cuyo estado se describe mediante el 3-qubit

$$\mathbf{q} = a_0 \,|\, \mathbf{000} > + a_1 \,|\mathbf{001} > + a_2 \,|\, \mathbf{010} > + a_3 \,|\, \mathbf{011} > + a_4 \,|\, \mathbf{100} > + a_5 \,|\, \mathbf{101} > + a_6 \,|\, \mathbf{110} > + a_7 \,|\, \mathbf{111} >$$

Entonces

Si una vez realizada la medición de los dos primeros qubits del sistema, se obtiene que ambos están en estado 0, el 3-qubit evoluciona al siguiente estado:

$$\mathbf{q} = \frac{\mathbf{a_0} \,|\, \mathbf{0\,0\,0} > + \mathbf{a_1} \,|\, \mathbf{0\,0\,1} >}{\sqrt{|\,\mathbf{a_0}\,|^2 + |\,\mathbf{a_1}\,|^2}}$$

Expresiones similares se obtienen para las restantes mediciones.

Los siguientes ejemplos muestran cómo, en caso de haber entrelazamiento, al medir el estado de un qubit queda determinado automáticamente el estado del otro qubit del sistema.

Ejemplo 2.5

Sea el sistema no entrelazado presentado en el Ejemplo 2.4, determinado por el 2-qubit

$$q = \frac{1}{2} \left(|\, 0\,0 > - |\, 0\,1 > + |\, 1\,0 > - |\, 1\,1 > \right)$$

Si al medir el primer qubit del sistema se observa que éste está en estado 0, de la Proposición 2.3 resulta que el 2-qubit evoluciona al estado

$$q = \frac{1}{\sqrt{2}} \,|\, 0\,0 > - \frac{1}{\sqrt{2}} \,|\, 0\,1 >$$

En este caso, de acuerdo a la Proposición 2.2, las probabilidades de medida sobre el segundo qubit son

$$P (X_2 = 0) = P (X_2 = 1) = \frac{1}{2}$$

Por otro lado, si al medir el primer qubit resulta que éste se encuentra en estado 1, de la Proposición 2.3 resulta que el 2-qubit evoluciona al estado

$$q = \frac{1}{\sqrt{2}} \; | 1\ 0 > - \frac{1}{\sqrt{2}} \; | 1\ 1 >$$

Nuevamente, al aplicar la Proposición 2.2 se obtiene que las probabilidades de medida sobre el segundo qubit son

$$P (X_2 = 0) = P (X_2 = 1) = \frac{1}{2}$$

En ambos casos se puede observar que la medida del primer qubit no afecta la medida del segundo qubit del sistema. ■

Ejemplo 2.6

Sea el sistema entrelazado presentado en el Ejemplo 2.3, determinado por el 2-qubit

$$q = \frac{1}{\sqrt{2}} \; | 0\ 1 > + \frac{1}{\sqrt{2}} \; | 1\ 0 >$$

Si al medir el primer qubit del sistema se observa que éste está en estado 0, de la Proposición 2.3 resulta que el 2-qubit evoluciona al estado

$$q = \; | 0\ 1 >$$

En este caso, de acuerdo a la Proposición 2.2, las probabilidades de medida sobre el segundo qubit son

$$P (X_2 = 0) = 0 \quad y \quad P (X_2 = 1) = 1$$

Esto significa que si al observar el primer electrón del sistema, éste se encuentra spin arriba (estado 0) entonces hay un 100 % de probabilidad de que el segundo electrón del sistema se halle spin abajo (estado 1).
Por otro lado, si al medir el primer qubit resulta que éste se encuentra en estado 1, de la Proposición 2.3 resulta que el 2-qubit evoluciona al estado

$$q = | 1\ 0 >$$

Nuevamente, al aplicar la Proposición 2.2 se obtiene que las probabilidades de medida sobre el segundo qubit son

$$P (X_2 = 0) = 1 \quad y \quad P (X_2 = 1) = 0$$

Lo cual significa que si al observar el primer electrón del sistema, éste se encuentra spin abajo (estado 1) entonces hay un 100 % de probabilidad de que el segundo electrón del sistema esté spin arriba (estado 0).
En ambos casos se puede observar que la medida del primer qubit sí afecta la medida del segundo qubit del sistema. ■

Ejemplo 2.7

Si se considera el sistema definido por el 2-qubit

$$q = \frac{1}{\sqrt{2}} \; | 0\ 0 > + \frac{1}{\sqrt{2}} \; | 1\ 1 >$$

Si al medir el primer qubit del sistema se observa que éste está en estado 0, de la Proposición 2.3 resulta que el 2-qubit evoluciona al estado

$$q = |0\,0\rangle$$

En este caso, de acuerdo a la Proposición 2.2, las probabilidades de medida sobre el segundo qubit son

$$P(X_2 = 0) = 1 \quad y \quad P(X_2 = 1) = 0$$

Esto significa que si al observar el primer electrón del sistema, éste se encuentra spin arriba (estado 0) entonces hay un 100 % de probabilidad de que el segundo electrón del sistema se halle también spin arriba (estado 0). ■

Como se puede comprobar, cuando dos partículas subatómicas están entrelazadas, el estado de una de ellas afecta sobre el estado de la otra, aunque se encuentren arbitrariamente alejadas entre sí y sin ningún medio físico que las una.
El estado de entrelazamiento entre dos o más qubits puede utilizarse para realizar la ***teleportación cuántica***, tema que se verá en el capítulo siguiente.
Se finalizará este capítulo mencionando las diferencias básicas entre un bit y un qubit.

2.4 Diferencias entre bits y qubits

El siguiente cuadro resume las principales diferencias entre un bit y un qubit.

Bit	***Qubit***
Es la unidad mínima de información en una computadora clásica.	Es la unidad mínima de información en una computadora cuántica.
Sólo puede almacenar un valor: 0 ó 1. Este hecho permite realizar cómputos de a un valor por vez.	Puede almacenar los valores 0 y 1 al mismo tiempo (Principio de Superposición). Este hecho permite realizar cómputos sobre ambos valores a la vez.
Una computadora clásica que opere con n bits, puede almacenar y procesar n bits al mismo tiempo.	Una computadora cuántica que opere con n qubits, puede almacenar y procesar al mismo tiempo 2^n bits (pues un qubit puede estar en estado 0 y 1 al mismo tiempo).
Se puede medir el valor almacenado en un bit, sin que se altere luego su valor.	En el momento en que el qubit es medido, éste colapsa en valor 0 o bien en valor 1.
Para transmitir el valor almacenado en un bit, es necesario que haya una conexión física entre el bit emisor y el bit receptor.	Debido a la Propiedad de Entrelazamiento, para transmitir el valor almacenado en un qubit, no es necesario que exista una conexión física entre el qubit emisor y el receptor.

Capítulo 3

Compuertas y circuitos cuánticos

Capítulo 3: Compuertas y circuitos cuánticos

En Mecánica Cuántica, una compuerta cuántica es un proceso físico capaz de transformar o de hacer evolucionar un sistema cuántico. En particular, en Computación Cuántica, una compuerta es un dispositivo electro-magnético con n qubits de entrada y n qubits de salida, que permite modificar el estado de los qubits que ingresan. Es equivalente a la compuerta lógica de los ordenadores convencionales.

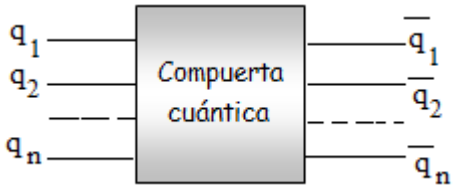


Figura 3.1: Representación gráfica de una compuerta cuántica. $\overline{q}_i$ *es el estado en que queda el qubit* q_i *al atravesar la compuerta, para i = 1, 2,…,n.*

Para modelar matemáticamente una compuerta cuántica, se acudirá al Segundo Postulado de la Mecánica Cuántica:

Segundo Postulado de la Mecánica ***Cuántica:***

La evolución de un sistema cuántico ***cerrado*** *(totalmente aislado, sin intercambio de energía con su medio ambiente) se describe mediante una transformación unitaria. Es decir que, si* q_1 *es el estado del sistema en el instante* t_1 *y* q_2 *es el estado en el instante* t_2*, entonces existe un operador unitario U de modo tal que* $q_2 = U\, q_1$*.*

Debido a que una compuerta cuántica de n qubits es capaz de modificar o hacer evolucionar el estado de los qubits del sistema, el Segundo Postulado sugiere asociar, a cada compuerta cuántica, una transformación unitaria definida en el espacio de los n-qubits, ya que éstos describen el estado del sistema. Por otro lado, como los n-qubits se identifican con vectores en $\mathbb{C}^{2^n}$, la transformación unitaria quedará definida en $\mathbb{C}^{2^n}$. Lo anteriormente expuesto se puede resumir en el siguiente Corolario:

Corolario 3.1: Representación matricial de una compuerta cuántica

Toda compuerta cuántica de n qubits se representa mediante una matriz unitaria U de orden 2^n*, es decir, una matriz* $U \in \mathbb{C}^{2^n} \mathbf{x} \mathbb{C}^{2^n}$ *que verifica*

$$U\,U^{+} = U^{+}\,U = I$$

Donde U^{+} *es la traspuesta conjugada de U e I es la matriz Identidad. La matriz unitaria U opera sobre un n-qubit.*

A lo largo de este capítulo se verá cómo se define la matriz unitaria asociada a una compuerta cuántica y recíprocamente, cómo una matriz unitaria genera una compuerta cuántica.
La siguiente Proposición prueba que una matriz unitaria preserva el módulo de un vector y, en consecuencia, transforma un n-qubit en otro n-qubit.

Proposición 3.1
Sea $v \in \mathbb{C}^m$ *y* U *una matriz unitaria m x m. Entonces si* $\| v \| = 1 \Rightarrow \| U v \| = 1$

Demostración

Si $\mathbf{v = (v_1, v_2, \dots , v_m)^T}$ y $\mathbf{w = (w_1, w_2, \dots, w_m)^T}$ entonces $< v ,w > = v^T \overline{w}$.

Por lo tanto

$\| U v \|^2 = < Uv, Uv > = (U v)^T \overline{U v} = v^T U^T \overline{U} \overline{v} = v^T \overline{\overline{U^T}\ \overline{U}}\ \overline{v} = v^T \overline{\overline{U^T}\ U}\ \overline{v} =$

$v^T \overline{U^+ U}\ \overline{v} = v^T \bar{I}\ \overline{v} = v^T I \overline{v} = v^T \overline{v} = < v , v > = \| v \|^2$
En consecuencia, $\| U v \|^2 = \| v \|^2$. De aquí resulta

$\| v \| = 1 \Rightarrow \| v \|^2 = 1 \Rightarrow \| U v \|^2 = 1 \Rightarrow \| U v \| = 1$ ■

Para cualquier cómputo cuántico se necesita diseñar un circuito cuántico, el cual se define de la siguiente manera:

Definición 3.2: Circuito cuántico

Un circuito cuántico es un dispositivo que consta de los siguientes elementos:

1) ***Uno o más qubits de entrada***. Éstos sólo pueden estar inicializados en $| 0 >$ o en $| 1 >$ ya que el electrón, al interactuar con el medio ambiente, puede permanecer únicamente en dos estados: spin arriba o spin abajo.

2) ***Una o más compuertas cuánticas***. Éstas permiten modificar el estado de uno o más qubits que componen el sistema.

3) ***Uno o más qubits de salida***. Ellos son el resultado de los cómputos realizados por las compuertas cuánticas. La medición final se efectúa sobre algunos de los qubits de salida.

El proceso de medición convierte el estado de un qubit $q = \alpha | 0 > + \beta | 1 >$ en un "bit clásico", que será 0 con una probabilidad $| \alpha |^2$ o bien 1 con una probabilidad $| \beta |^2$. Gráficamente, el procedimiento de medición de un qubit se lo simboliza así:

Figura 3.2: Representación gráfica del proceso de medición de un qubit q. Las líneas dobles se utilizan para simbolizar los bits clásicos: 0 y 1.

El siguiente gráfico muestra un ejemplo de circuito cuántico formado por tres qubits de entrada y tres de salida:

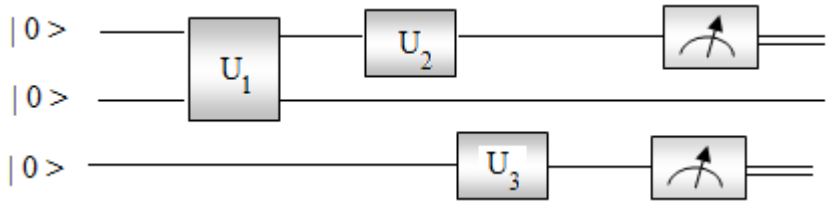


Figura 3.3: Un ejemplo de circuito cuántico.

Las compuertas cuánticas más utilizadas son las de un qubit y dos qubits. Ello se debe a que cualquier circuito cuántico puede diseñarse empleando solamente este tipo de compuertas. Por esta razón, en la siguiente sección, se las describirá detalladamente.

3.1 Compuertas cuánticas de un qubit

Desde el punto de vista físico, las compuertas cuánticas que operan sobre un qubit son dispositivos que modifican el spin del electrón. Esto se logra generando campos magnéticos a lo largo de diferentes direcciones, los cuales permiten cambiar la orientación del spin del electrón. Pero desde el punto de vista matemático, según el Corolario 3.1, las compuertas cuánticas que operan sobre un qubit se representan mediante matrices unitarias de orden 2. La siguiente Proposición muestra cómo es la representación general de este tipo de matrices:

Proposición 3.2

Si U es una matriz unitaria 2 x 2 entonces

$$U = \begin{pmatrix} \cos(\alpha)\, e^{i\,\theta_1} & \operatorname{sen}(\alpha)\, e^{i\,\theta_2} \\ \operatorname{sen}(\alpha)\, e^{i\,\theta_3} & -\cos(\alpha)\, e^{i\,(-\theta_1+\theta_2+\theta_3)} \end{pmatrix}$$

Donde

$$0 \le \alpha \le \frac{\pi}{2}\ ,\quad 0 \le \theta_1\ ,\ \theta_2\ ,\ \theta_3 < 2\,\pi$$

Demostración

Sea $U = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$. Si se expresan los coeficientes de U en forma exponencial resulta

$$\begin{cases} a = \rho_1\, e^{i\,\theta_1} \\ b = \rho_2\, e^{i\,\theta_2} \\ c = \rho_3\, e^{i\,\theta_3} \\ d = \rho_4\, e^{i\,\theta_4} \end{cases} \qquad [1]$$

Con $\rho_1, \rho_2, \rho_1, \rho_4 \geq 0 \ \wedge \ 0 \leq \theta_1, \theta_2, \theta_3, \theta_4 < 2\pi$

Por otro lado, como $U\,U^{+} = I$ se obtiene

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \cdot \begin{pmatrix} \bar{a} & \bar{c} \\ \bar{b} & \bar{d} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \Rightarrow \begin{cases} a\,\bar{a} + b\,\bar{b} = 1 \\ c\,\bar{c} + d\,\bar{d} = 1 \\ a\,\bar{c} + b\,\bar{d} = 0 \end{cases} \Rightarrow$$

$$\begin{cases} \rho_1^2 + \rho_2^2 = 1 & [2] \\ \rho_3^2 + \rho_4^2 = 1 & [3] \\ \rho_1\,\rho_3\, e^{i(\theta_1 - \theta_3)} + \rho_2\,\rho_4\, e^{i(\theta_2 - \theta_4)} = 0 & [4] \end{cases}$$

Al expresar el par ordenado (ρ_1, ρ_2) en coordenadas polares, se tiene

$$\begin{cases} \rho_1 = r\cos(\alpha) \\ \rho_2 = r\,\mathrm{sen}(\alpha) \end{cases}$$

Donde $0 \leq \alpha \leq \frac{\pi}{2}$

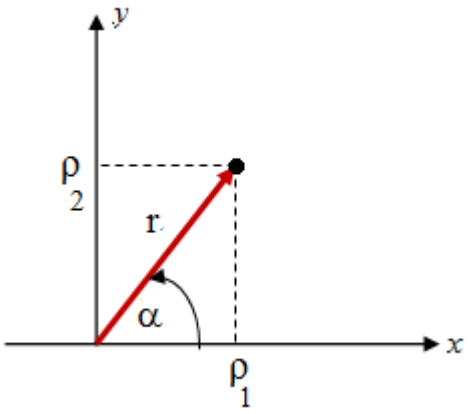


Figura 3.4: Representación de (ρ_1, ρ_2) *en coordenadas polares*

De la ecuación [2] se obtiene que r = 1 y, en consecuencia,

$$\begin{cases} \rho_1 = \cos(\alpha) \\ \rho_2 = \mathrm{sen}(\alpha) \end{cases} \qquad [5]$$

De manera similar, al expresar (ρ_3, ρ_4) en coordenadas polares, y haciendo uso de la ecuación [3] , se tiene

$$\begin{cases} \rho_3 = \cos(\beta) \\ \rho_4 = \mathrm{sen}(\beta) \end{cases} \qquad [6]$$

Con $0 \leq \beta \leq \frac{\pi}{2}$. Al sustituir las expresiones [5] y [6] en [4] resulta

$\cos(\alpha)\cos(\beta)\, e^{i(\theta_1 - \theta_3)} + \mathrm{sen}(\alpha)\,\mathrm{sen}(\beta)\ e^{i(\theta_2 - \theta_4)} = 0 \Rightarrow$

$$\cos(\alpha)\cos(\beta)\, e^{i(\theta_1-\theta_3)} = -\operatorname{sen}(\alpha)\operatorname{sen}(\beta)\, e^{i(\theta_2-\theta_4)} \Rightarrow$$

$$\left|\cos(\alpha)\cos(\beta)\, e^{i(\theta_1-\theta_3)}\right| = \left|-\operatorname{sen}(\alpha)\operatorname{sen}(\beta)\, e^{i(\theta_2-\theta_4)}\right| \Rightarrow$$

$$\cos(\alpha)\cos(\beta) = \operatorname{sen}(\alpha)\operatorname{sen}(\beta) \Rightarrow \operatorname{tg}(\beta) = \operatorname{cotg}(\alpha)$$

Al aplicar esta última ecuación y, teniendo en cuenta las siguientes identidades trigonométricas:

$$\operatorname{sen}(\beta) = \frac{\operatorname{tg}(\beta)}{\sqrt{1+\operatorname{tg}^2(\beta)}} \quad , \quad \cos(\beta) = \frac{1}{\sqrt{1+\operatorname{tg}^2(\beta)}}$$

Se tiene que

$$\begin{cases} \operatorname{sen}(\beta) = \cos(\alpha) \\ \cos(\beta) = \operatorname{sen}(\alpha) \end{cases} \qquad [7]$$

Luego, al reemplazar [7] en [6] se obtiene

$$\begin{cases} \rho_3 = \operatorname{sen}(\alpha) \\ \rho_4 = \cos(\alpha) \end{cases} \qquad [8]$$

Sustituyendo [8] en [4] resulta

$$\cos(\alpha)\operatorname{sen}(\alpha)\, e^{i(\theta_1-\theta_3)} + \operatorname{sen}(\alpha)\cos(\alpha)\, e^{i(\theta_2-\theta_4)} = 0 \Rightarrow$$

$$\cos(\alpha)\operatorname{sen}(\alpha)\left(e^{i(\theta_1-\theta_3)} + e^{i(\theta_2-\theta_4)}\right) = 0$$

Como esta ecuación debe ser válida $\forall\ \alpha$, con $0 \le \alpha \le \frac{\pi}{2}$ entonces

$$\left(e^{i(\theta_1-\theta_3)} + e^{i(\theta_2-\theta_4)}\right) = 0 \Rightarrow e^{i(\theta_2-\theta_4)} = -e^{i(\theta_1-\theta_3)} \Rightarrow$$

$$e^{i\theta_4} = -e^{i(-\theta_1+\theta_2+\theta_3)} \qquad [9]$$

Reemplazando [5] , [8] y [9] en [1] se obtiene

$$\begin{cases} a = \cos(\alpha)\, e^{i\theta_1} \\ b = \operatorname{sen}(\alpha)\, e^{i\theta_2} \\ c = \operatorname{sen}(\alpha)\, e^{i\theta_3} \\ d = -\cos(\alpha)\, e^{i(-\theta_1+\theta_2+\theta_3)} \end{cases}$$

En consecuencia,

$$U = \begin{pmatrix} \cos(\alpha)\, e^{i\theta_1} & \operatorname{sen}(\alpha)\, e^{i\theta_2} \\ \operatorname{sen}(\alpha)\, e^{i\theta_3} & -\cos(\alpha)\, e^{i(-\theta_1+\theta_2+\theta_3)} \end{pmatrix} \quad \blacksquare$$

Teóricamente, cualquier matriz unitaria de orden 2 define una compuerta cuántica de un qubit. Sin embargo, las más utilizadas se presentarán a continuación.

Principales compuertas cuánticas de un qubit

En esta sección se definirán las principales compuertas cuánticas de un qubit en términos de sus matrices unitarias asociadas.

Las matrices de Pauli

Las matrices de Pauli son tres: X, Y , Z donde

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} , \; Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} , \; Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Para analizar el efecto que tienen las compuertas asociadas a estas matrices sobre un qubit, basta hacerlo sobre los qubits $| 0 >$ y $| 1 >$, ya que éstos constituyen una base para el sistema formado por un qubit. En particular, el resultado que produce la compuerta X es:

$$X | 0 > = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = | 1 >$$

$$X | 1 > = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = | 0 >$$

De este modo, esta compuerta cambia el estado 0 al estado 1 y el estado 1 al estado 0 . Por esta razón a la matriz de Pauli X se la suele llamar “la compuerta NOT” Físicamente significa que, si el electrón estaba “spin arriba”, al atravesar la compuerta X pasa a estar “spin abajo” y viceversa.

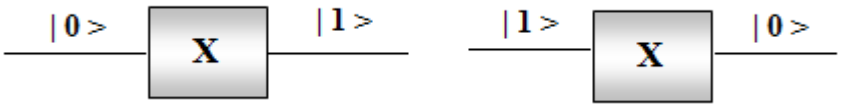


Figura 3.5: Efecto de la matriz X sobre los qubits de la base.

Por otro lado, el efecto de la matriz Z sobre los qubits de la base es el siguiente:

$$Z | 0 > = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = | 0 >$$

$$Z | 1 > = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ -1 \end{pmatrix} = - | 1 >$$

Es decir que, al estado 0 lo deja inalterado y al estado 1 le cambia el signo.

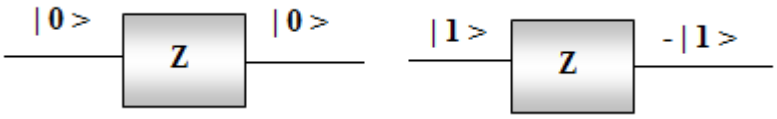


Figura 3.6: Efecto de la matriz Z sobre los qubits de la base.

La compuerta de Hadamard

Esta compuerta es sumamente relevante y lleva este nombre en honor al matemático francés Jacques Hadamard (1865-1963). Su matriz asociada es la siguiente:

$$H = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix}$$

A continuación se muestra cómo actúa esta matriz sobre los qubits de la base:

$$H\,|\,0\,> = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} = \frac{1}{\sqrt{2}}\,|\,0\,> + \frac{1}{\sqrt{2}}\,|\,1\,>$$

$$H\,|\,1\,> = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{pmatrix} = \frac{1}{\sqrt{2}}\,|\,0\,> - \frac{1}{\sqrt{2}}\,|\,1\,>$$

Como se puede comprobar, la compuerta de Hadamard transforma los qubits | 0 > y | 1 > en estados de superposición, permitiendo estar en estado 0 y 1 al mismo tiempo. Por otro lado, los qubits transformados H | 0 > y H | 1 > tienen la misma probabilidad de que colapsen a estado 0 o a estado 1 en el instante en que entren en contacto con el medio ambiente (es decir, en el momento en que sean observados o sean sometidos a una medición). El valor de dicha probabilidad es $\left(\frac{1}{\sqrt{2}}\right)^2 = \frac{1}{2}$.

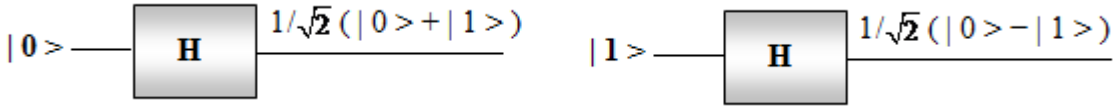


Figura 3.7: Efecto de la compuerta H sobre los qubits de la base.

Físicamente la superposición se logra creando una “trampa de iones”. Sin embargo, es muy difícil mantenerla en el tiempo, ya que el electrón debe estar completamente aislado, porque si interactúa con el medio ambiente, el estado de superposición se destruye. Esta es una de las causas principales por la cual se torna muy dificultosa la construcción de computadoras cuánticas.

La compuerta R o de desplazamiento de fase

Dado un ángulo θ expresado en radianes, la compuerta de deslazamiento de fase R_θ tiene la siguiente forma matricial:

$$R_\theta = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\,\theta} \end{pmatrix}$$

El efecto de esta compuerta sobre los qubits de la base es el siguiente:

$$R_\theta \mid 0 > = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\,\theta} \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \mid 0 >$$

$$R_\theta \mid 1 > = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\,\theta} \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ e^{i\,\theta} \end{pmatrix} = e^{i\,\theta} \mid 1 >$$

Es decir que al qubit $\mid 0 >$ lo deja intacto y a $\mid 1 >$ lo transforma en $e^{i\,\theta} \mid 1 >$.
La compuerta R_θ, junto con la de Hadamard, serán utilizadas para diseñar el circuito cuántico para la Transformada Cuántica de Fourier.

Las matrices de Pauli cobran importancia debido a que cualquier matriz unitaria de orden 2 puede escribirse, junto con la matriz Identidad, como combinación lineal de estas matrices. Es decir, cualquier compuerta para un qubit puede generarse a partir de la matriz Identidad y de las matrices de Pauli, como se demostrará en la siguiente Proposición:

Proposición 3.3

Si U es una matriz unitaria 2 x 2 entonces U se escribe como combinación lineal de las matrices de Pauli y de la matriz Identidad.

Demostración

Sea U una matriz unitaria 2x2. Por la Proposición 3.2 se sabe que

$$U = \begin{pmatrix} \cos(\alpha)\, e^{i\,\theta_1} & \operatorname{sen}(\alpha)\, e^{i\,\theta_2} \\ \operatorname{sen}(\alpha)\, e^{i\,\theta_3} & -\cos(\alpha)\, e^{i\,(-\theta_1+\theta_2+\theta_3)} \end{pmatrix}$$

Se tratará de hallar a, b, c, y d de modo tal que

$$U = a\,I + b\,X + c\,Y + d\,Z$$

Es decir,

$$\begin{pmatrix} \cos(\alpha)\, e^{i\,\theta_1} & \operatorname{sen}(\alpha)\, e^{i\,\theta_2} \\ \operatorname{sen}(\alpha)\, e^{i\,\theta_3} & -\cos(\alpha)\, e^{i\,(-\theta_1+\theta_2+\theta_3)} \end{pmatrix} = a \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + b \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} + c \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} + d \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Luego,

$$\begin{pmatrix} \cos(\alpha)\ e^{i\,\theta_1} & \operatorname{sen}(\alpha)\ e^{i\,\theta_2} \\ \operatorname{sen}(\alpha)\ e^{i\,\theta_3} & -\cos(\alpha)\ e^{i\,(-\theta_1+\theta_2+\theta_3)} \end{pmatrix} = \begin{pmatrix} a+d & b-i\,c \\ b+i\,c & a-d \end{pmatrix}$$

Al igualar los coeficientes de ambas matrices resulta el siguiente sistema de ecuaciones:

$$\begin{cases} a+d=\cos(\alpha)\ e^{i\,\theta_1} \\ b-i\,c=\operatorname{sen}(\alpha)\ e^{i\,\theta_2} \\ b+i\,c=\operatorname{sen}(\alpha)\ e^{i\,\theta_3} \\ a-d=-\cos(\alpha)\ e^{i\,(-\theta_1+\theta_2+\theta_3)} \end{cases}$$

Cuya solución es:

$$\begin{cases} a=\dfrac{\cos(\alpha)}{2}\left(e^{i\,\theta_1}-e^{i\,(-\theta_1+\theta_2+\theta_3)}\right) \\ b=\dfrac{\operatorname{sen}(\alpha)}{2}\left(e^{i\,\theta_2}+e^{i\,\theta_3}\right) \\ c=\dfrac{\operatorname{sen}(\alpha)}{2\,i}\left(-e^{i\,\theta_2}+e^{i\,\theta_3}\right) \\ d=\dfrac{\cos(\alpha)}{2}\left(e^{i\,\theta_1}+e^{i\,(-\theta_1+\theta_2+\theta_3)}\right) \end{cases}$$

■

Pero cualquier matriz unitaria de orden 2 también puede escribirse como producto de matrices de rotación, según se afirma en la siguiente Proposición:

Proposición 3.4

Si U es una matriz unitaria 2 x 2 entonces

$$U=\begin{pmatrix} e^{i\left(\frac{\theta_2+\theta_3}{2}\right)} & 0 \\ 0 & e^{i\left(\frac{\theta_2+\theta_3}{2}\right)} \end{pmatrix}\begin{pmatrix} e^{i\left(\frac{\theta_1-\theta_3}{2}\right)} & 0 \\ 0 & e^{-i\left(\frac{\theta_1-\theta_3}{2}\right)} \end{pmatrix}\begin{pmatrix} \cos(\alpha) & \operatorname{sen}(\alpha) \\ \operatorname{sen}(\alpha) & -\cos(\alpha) \end{pmatrix}\begin{pmatrix} e^{i\left(\frac{\theta_1-\theta_2}{2}\right)} & 0 \\ 0 & e^{-i\left(\frac{\theta_1-\theta_2}{2}\right)} \end{pmatrix}$$

Demostración

Es consecuencia de la Proposición 3.2 y de desarrollar el producto matricial. ■

La Esfera de Bloch y las compuertas cuánticas de un qubit

La Esfera de Bloch resulta sumamente útil para visualizar el efecto que producen las compuertas cuánticas sobre un qubit. La siguiente simulación, hecha con el software Wolfram Mathematica 9.0, muestra cómo un qubit q es transformado por una matriz unitaria U. El código fuente se muestra a continuación:

```
Interpretation[{α = 1, β = 0, a = 1/√2 , b = 1/√2 , c = 1/√2 , d = - 1/√2 },

 Panel[Grid[{{Style["Ingrese el valor del qubit q= (α/β)  con α, β ∈ ℂ  y  |α|^2+ |β|^2= 1 ", Bold], SpanFromLeft},
    {"q=", MatrixForm[{InputField[Dynamic[α], FieldSize → 4], InputField[Dynamic[β], FieldSize → 4]}]},
    {Style["Ingrese los coeficientes de la compuerta U  (U es una matriz unitaria)", Bold], SpanFromLeft},
    {"U=", MatrixForm[{{InputField[Dynamic[a], FieldSize → 4], InputField[Dynamic[b], FieldSize → 4]},
       {InputField[Dynamic[c], FieldSize → 4], InputField[Dynamic[d], FieldSize → 4] } } ]} }] ],

H = {{a, b}, {c, d}};
tita1 = Arg[α];
tita2 = Arg[β];
ro1 = Abs[α];
ro2 = Abs[β];
fi = If[ro1 ≠ 0, 2 ArcTan[ro2/ro1] , 2 Pi / 2];
tita = tita2 - tita1;
q = { Sin[fi] Cos[tita], Sin[ fi] Sin[tita], Cos[fi] };
Punto = Graphics3D[{Yellow, Specularity[White, 5], Sphere[q, 0.06] }];

αt = Extract[H.{α, β}, {1}];
βt = Extract[H.{α, β}, {2}];
tita11 = Arg[αt];
tita22 = Arg[βt];
ro11 = Abs[αt];
ro22 = Abs[βt];
```

```
fit = If[ro11 ≠ 0, 2 ArcTan[ro22/ro11] , 2 Pi / 2];
titat = tita22 - tita11;
Uq = { Sin[fit] Cos[titat], Sin[ fit] Sin[titat], Cos[fit] };

Esfera = ParametricPlot3D[{ Sin[ϕ] Cos[θ], Sin[ ϕ] Sin[θ], Cos[ϕ] }, { ϕ, 0, Pi}, { θ, 0, 2 Pi} , Boxed → False, Axes → True, AxesOrigin → {0, 0, 0}
  PlotRange → {{-1.3, 1.3}, {-1.3, 1.3}, {-1.3, 1.3}}, ImageSize → {500, 450}, Background → Black];
Ejez = Graphics3D[{White, Arrowheads[0.18], Arrow[Tube[{{0, 0, -2.5}, {0, 0, 2.5}}]]}];


Manipulate[
 Curva = ParametricPlot3D[If[ s <= 1 , { Sin[fi] Cos[tita + (titat - tita) s], Sin[ fi] Sin[tita + (titat - tita) s], Cos[fi] } ,
    { Sin[fi + (fit - fi) (s - 1)] Cos[titat], Sin[ fi + (fit - fi) (s - 1)] Sin[titat], Cos[fi + (fit - fi) (s - 1)] }], {s, 0, t},
   PlotStyle → {Black, Thickness[0.005]}];
 Pt = If[ t <= 1 , { Sin[fi] Cos[tita + (titat - tita) t], Sin[ fi] Sin[tita + (titat - tita) t], Cos[fi] } ,
   { Sin[fi + (fit - fi) (t - 1)] Cos[titat], Sin[ fi + (fit - fi) (t - 1)] Sin[titat], Cos[fi + (fit - fi) (t - 1)] }];
 Puntot = Graphics3D[{Blue, Specularity[White, 5], Sphere[Pt, 0.06] }];
 Ejev = Graphics3D[{Red, Arrowheads[0.04], Arrow[Tube[{Pt, Pt + 0.35 Pt/Norm[Pt]}, 0.02]] }]; Show[Esfera, Curva, Puntot, Punto, Ejez, Ejev],
 Panel["                                LA ESFERA DE BLOCH                                "],
 Panel["Representacion del qubit q en la esfera de Bloch:"], {q}, Panel["Representacion del qubit U.q en la esfera de Bloch:"],
 {Uq}, {{t, 0.00001, "Transformacion del qubit q al aplicar la compuerta U"}, 0.00001, 2, ControlType → Trigger }, SaveDefinitions -> True]
]
```

Figura 3.8: Código fuente de la esfera de Bloch para visualizar el efecto de una compuerta cuántica.

Para ejecutar la simulación se deberá ingresar las componentes del qubit q y los coeficientes de la matriz unitaria U:

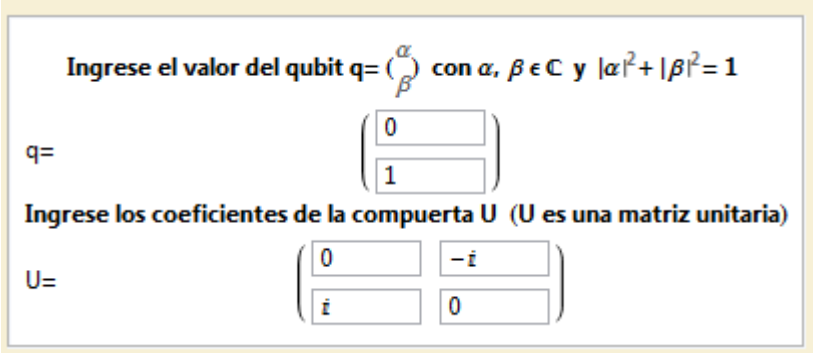


Figura 3.9: Datos a ingresar para ejecutar la simulación: las componentes del qubit q y los coeficientes de la matriz U.

El siguiente gráfico muestra la simulación resultante:

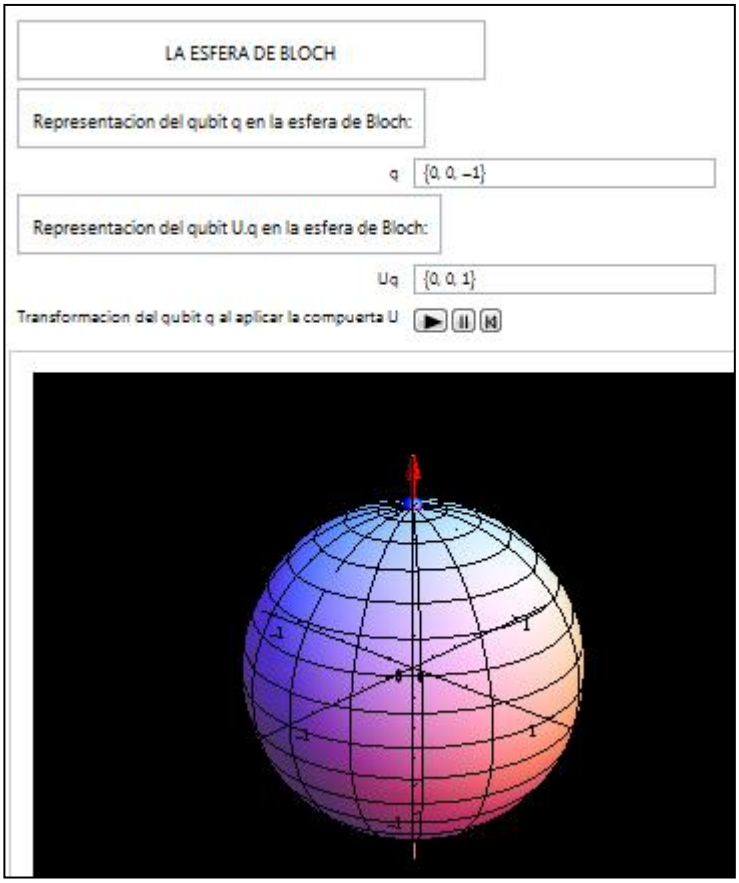


Figura 3.10: Simulación del efecto de una compuerta cuántica.

Al ejecutar la simulación se puede visualizar claramente cómo va cambiando la posición del eje de rotación del electrón, por la acción de una determinada compuerta. Por ejemplo, la compuerta de Hadamard actúa sobre un qubit produciendo una rotación de 90° sobre el eje y , seguida de una rotación de 180° sobre el eje x.

Por otro lado, si un qubit atraviesa la compuerta R_θ, sobre la esfera de Bloch se verá que el qubit se desplazará θ radianes a lo largo de su propia latitud.

3.2 Compuertas cuánticas de dos qubits

Nuevamente, según el Corolario 3.1, las compuertas cuánticas que operan sobre un sistema de dos qubits se representan mediante matrices unitarias 4 x 4, las cuales operan sobre un 2-qubit. Como se sabe, una base ortonormal para el espacio de los 2-qubits es

$$\{ | 0\,0 >, | 0\,1 >, | 1\,0 >, | 1\,1 > \}$$

Donde

$$| 0\,0 > = | 0 > \otimes | 0 > = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} \approx \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$| 0\,1 > = | 0 > \otimes | 1 > = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} \approx \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

$$| 1\,0 > = | 1 > \otimes | 0 > = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} \approx \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

$$| 1\,1 > = | 1 > \otimes | 1 > = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} \approx \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Luego, para definir la matriz unitaria U asociada a una compuerta cuántica, bastará con definir la transformación U sobre cada 2-qubit de la base, expresándola como combinación lineal de los elementos de dicha base. Es decir, se especificará:

$$U| 0\,0 > = a_{11} | 0\,0 > + a_{12} | 0\,1 > + a_{13} | 1\,0 > + a_{14} | 1\,1 >$$

$$U| 0\,1 > = a_{21} | 0\,0 > + a_{22} | 0\,1 > + a_{23} | 1\,0 > + a_{24} | 1\,1 >$$

$$U| 1\,0 > = a_{31} | 0\,0 > + a_{32} | 0\,1 > + a_{33} | 1\,0 > + a_{34} | 1\,1 >$$

$$U| 1\,1 > = a_{41} | 0\,0 > + a_{42} | 0\,1 > + a_{43} | 1\,0 > + a_{44} | 1\,1 >$$

De este modo, la matriz unitaria U será

$$U = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{pmatrix}$$

A continuación se describirán las principales compuertas cuánticas aplicables a un sistema de dos qubits.

Principales compuertas cuánticas de dos qubits

La compuerta Swap

Esta compuerta intercambia dos qubits. Su matriz asociada se la simboliza por Uswap y se la define por Uswap | x y > = | y x > con x, y $\in \{0,1\}$. De este modo, el efecto de esta compuerta sobre cada 2-qubit de la base es:

$$\text{Uswap} \mid 0\ 0 > = \mid 0\ 0 > = 1.\ \mid 0\ 0 > + 0 . \mid 0\ 1 > + 0. \mid 1\ 0 > + 0. \mid 1\ 1 >$$

$$\text{Uswap}\mid 0\ 1 > = \mid 1\ 0 > = 0 . \mid 0\ 0 > + 0 . \mid 0\ 1 > + 1. \mid 1\ 0 > + 0. \mid 1\ 1 >$$

$$\text{Uswap}\mid 1\ 0 > = \mid 0\ 1 > = 0.\ \mid 0\ 0 > + 1 . \mid 0\ 1 > + 0. \mid 1\ 0 > + 0. \mid 1\ 1 >$$

$$\text{Uswap}\mid 1\ 1 > = \mid 1\ 1 > = 0.\ \mid 0\ 0 > + 0 . \mid 0\ 1 > + 0. \mid 1\ 0 > + 1. \mid 1\ 1 >$$

Por lo tanto, la matriz asociada a esta compuerta es

$$\text{Uswap} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

La representación gráfica de esta compuerta es la siguiente:

Figura 3.11: Representación gráfica de la compuerta Swap.

La compuerta Xor

Antes de describir esta compuerta, y para comprender porqué lleva este nombre, se recordará que el operador lógico Xor, también llamado "O excluyente", tiene la siguiente tabla de valores:

x	y	x Xor y
0	0	0
0	1	1
1	0	1
1	1	0

Como se puede comprobar, este operador lógico coincide con la "suma módulo 2", que se representará con el símbolo $\oplus$.

La compuerta cuántica Xor se la define a través de la transformación unitaria Uxor | x y > = | x , x ⊕ y > Por lo tanto

$$\text{Uxor} \mid 0\ 0 > = \mid 0, 0 \oplus 0 > = \mid 0\ 0 >$$

$$\text{Uxor} \mid 0\ 1 > = \mid 0, 0 \oplus 1 > = \mid 0\ 1 >$$

$$\text{Uxor} \mid 1\ 0 > = \mid 1 , 1 \oplus 0 > = \mid 1\ 1 >$$

$$\text{Uxor} \mid 1\ 1 > = \mid 1, 1 \oplus 1 > = \mid 1\ 0 >$$

Su expresión matricial es entonces

$$\text{Uxor} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Y su representación gráfica es la siguiente:

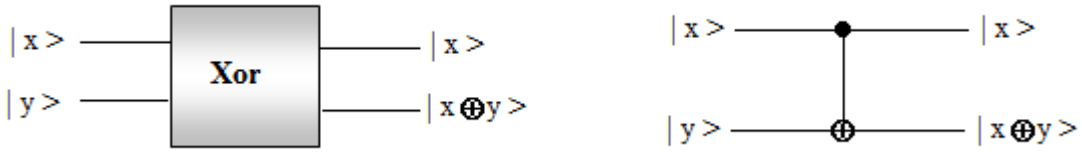


Figura 3.12: Dos representaciones gráficas de la compuerta Xor.

La compuerta Xor es de gran importancia ya que cualquier compuerta de n qubits puede ser construida mediante la compuerta Xor y compuertas de un qubit. La demostración de esta afirmación supera el objetivo de este trabajo. Sin embargo, a modo ilustrativo, en el siguiente ejemplo se mostrará cómo la compuerta Swap puede ser implementada mediante tres compuertas Xor.

Ejemplo 3.1

El siguiente circuito intercambia dos qubits por medio del uso de tres Xor cuánticos:

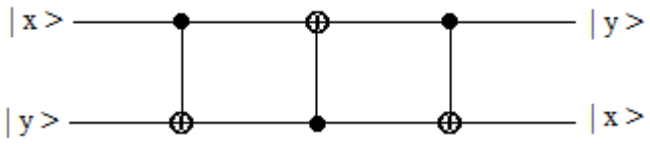


Figura 3.13: Circuito cuántico que intercambia dos qubits

Siguiendo el circuito de la Figura 3.11 de izquierda a derecha, se obtiene la transformación que intercambia dos qubits:

Resultado del primer Xor: $\mid x , y > \rightarrow \mid x , x \oplus y >$

Resultado del segundo Xor: $\mid (x \oplus y) \oplus x , x \oplus y > \equiv \mid y , x \oplus y >$

Resultado del tercer Xor: $\mid y , y \oplus (x \oplus y) > \equiv \mid y , x >$ ■

La compuerta f

A partir de una función booleana f: $\{0, 1\} \rightarrow \{0, 1\}$ se puede construir una compuerta cuántica de dos qubits, llamada la compuerta **f**, mediante una transformación unitaria U_f definida del siguiente modo:

$$\mathbf{U_f \,|\, x \;\; y \rangle = |\, x \,,\; y \oplus f(x) \rangle}$$

Donde $\oplus$ representa la suma módulo 2 y $\mathbf{x}, \mathbf{y} \in \{0, 1\}$.
Por lo tanto, la compuerta cuántica asociada a U_f opera del siguiente modo:

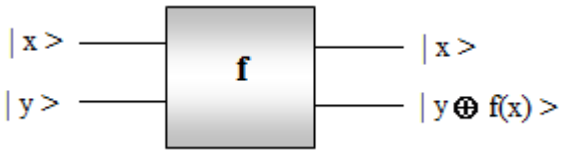


Figura 3.14: Efecto de la compuerta f.

Ejemplo 3.2

Dada f: $\{0, 1\} \rightarrow \{0, 1\}$ definida por f(0) = 1 , f(1) = 0, hallar la matriz asociada a la transformación U_f.

Solución

$U_f \,|\, 0\; 0 \rangle = |\, 0 \,,\; 0 \oplus f(0) \rangle = |\, 0 \,,\; 0 \oplus 1 \rangle = |\, 0\; 1 \rangle$

$U_f \,|\, 0\; 1 \rangle = |\, 0 \,,\; 1 \oplus f(0) \rangle = |\, 0 \,,\; 1 \oplus 1 \rangle = |\, 0\; 0 \rangle$

$U_f \,|\, 1\; 0 \rangle = |\, 1 \,,\; 0 \oplus f(1) \rangle = |\, 1 \,,\; 0 \oplus 0 \rangle = |\, 1\; 0 \rangle$

$U_f \,|\, 1\; 1 \rangle = |\, 1 \,,\; 1 \oplus f(1) \rangle = |\, 1 \,,\; 0 \oplus 1 \rangle = |\, 1\; 1 \rangle$

Luego, la matriz asociada es

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

■

De lo expuesto anteriormente se puede deducir que las compuertas cuánticas presentan cierta similitud con las compuertas lógicas. En la siguiente sección se describirán sus características principales.

3.3 Características de las compuertas cuánticas

Las compuertas cuánticas, materializadas en dispositivos de algún tipo, son las encargadas de realizar las operaciones, los cómputos. Tienen fundamentalmente las siguientes propiedades:

■ *Reversibilidad.*

■ *El número de entradas es igual al número de salidas.*

■ *Las operaciones aritméticas que realiza son de orden polinomial.*

Se verán detalladamente estas propiedades:

■ ***Reversibilidad***

Una compuerta es reversible si a partir de los datos de salida se pueden recuperar los datos de entrada. Para el caso de las compuertas cuánticas, esta propiedad está garantizada ya que éstas se implementan mediante operadores unitarios, los cuales permiten rescatar los datos de entrada:

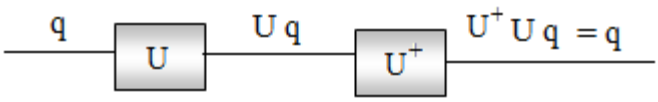


Figura 3.15: Reversibilidad de una compuerta cuántica. Al multiplicar U q por U^+, la traspuesta conjugada de la matriz U, se puede recuperar q.

El motivo por el cual una compuerta cuántica debe ser reversible se debe a que un mismo qubit debe ser capaz de efectuar múltiples operaciones. Esto sólo es posible si el qubit, una vez que realizó una determinada operación al atravesar una compuerta, recupera su estado anterior para así llevar a cabo la siguiente operación.

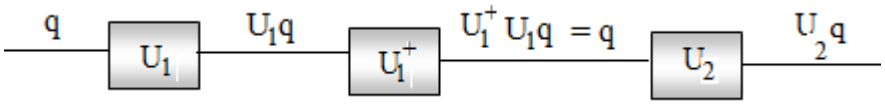


Figura 3.16: Si el qubit q efectuó una operación al atravesar la compuerta U_1, ésta debe ser reversible para poder efectuar la siguiente operación mediante la compuerta U_2.

Cabe preguntarse: ¿No se podría reproducir un qubit para que éste efectúe múltiples operaciones, sin necesidad de diseñar una compuerta reversible? La respuesta es No, y ello se debe a que "*no se puede clonar un qubit*". Este fenómeno recibe el nombre de "*Teorema de no clonación*", el cual será demostrado matemáticamente en la siguiente sección.

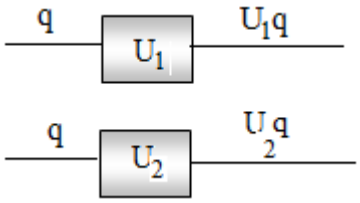


Figura 3.17: No es posible "clonar" un qubit q para que efectúe dos operaciones al atravesar las compuertas U_1 y U_2.

■ ***El número de entradas es igual al número de salidas***

Si la compuerta cuántica tuviera menos salidas que entradas, se desecharía información. Este hecho se traduce físicamente en una disipación de la energía que se transforma en calor y, por consiguiente, estos sistemas se calentarían. Como los estados cuánticos del átomo son extremadamente frágiles, la superposición podría verse afectada ante la más mínima fluctuación de la temperatura, lo que provocaría errores de cálculo.
Por otro lado, si la compuerta tuviera más salidas que entradas, la compuerta encargada de recuperar los datos de entrada tendría menos salidas que entradas. Pero, por lo visto anteriormente, se producirían errores de cálculo.
Por esta razón una compuerta cuántica debe tener el mismo número de entradas que de salidas.

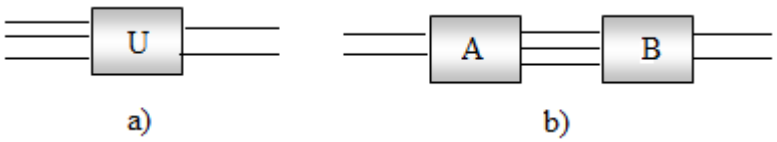


Figura 3.18: a) Compuerta con más entradas que salidas. b) Si la compuerta A tiene más salidas que entradas, la compuerta B, encargada de recuperar los datos de entrada, debería tener más entradas que salidas.

■ ***Las operaciones aritméticas que realiza son de orden polinomial.***

Si una compuerta cuántica tiene "x" entradas, el número de operaciones aritméticas (suma, resta, multiplicación o división) que realiza la compuerta es un polinomio en la variable "x". En consecuencia, estas operaciones las lleva a cabo en tiempo polinomial. En la siguiente Proposición se demuestra esta afirmación.

Proposición 3.5

Sea U la matriz unitaria asociada a una compuerta de n qubits. Si x representa el número de entradas de la compuerta, entonces el número de operaciones aritméticas que realiza la compuerta es del orden $O(x^2)$.

Demostración

La matriz unitaria U asociada a la compuerta de n qubits tiene 2^n filas y 2^n columnas y opera sobre un n-qubit. Por otro lado, como un n-qubit q se representa mediante un vector de 2^n componentes, la matriz U tendrá $x = 2^n$ entradas. Sea

$$\left\{ | 0 > , \ | 1 > , | 2 > , ... , \ | 2^n - 1 > \right\}$$

Una base para el espacio de los n-qubits**.** Entonces

$$q = \sum_{i=0}^{2^n - 1} a_i \ | i >$$

Donde

$$a_i \in \mathbb{C} \ \ \forall \ i = 0 ,..., 2^n - 1 \quad y \quad \sum_{i=0}^{2^n - 1} | a_i |^2 = 1$$

Se calculará el número de operaciones llevadas a cabo al calcular Uq.

$$Uq = U \left(\sum_{i=0}^{2^n - 1} a_i \ | i > \right) = \sum_{i=0}^{2^n - 1} a_i \ U \ | i >$$

1) Cálculo de U | i > , para i = 0, 1, …, 2^n -1
Para cada i, | i > es un vector de 2^n componentes, de las cuales sólo una es igual a 1 y las demás son nulas. Por lo tanto, U | i > da por resultado un vector columna de U, que tiene 2^n componentes. Luego, al calcular a_i U| i > se realizan 2^n multiplicaciones. En consecuencia, al calcular U | i > , para i = 0, 1, …, 2^n -1, se efectúan 2^n. 2^n multiplicaciones.

2) Cálculo de $\sum_{i=0}^{2^n-1} v_i$ donde $v_i = a_i \, U|\, i >$

Para cada i, v_i es un vector de 2^n componentes. Luego, al calcular dicha sumatoria se efectúan 2^n- 1 sumas de vectores de 2^n componentes. Por lo tanto, se realizan $(2^n-1)\, 2^n$ sumas.

De este modo, el total de operaciones aritméticas necesarias para calcular Uq es

$$2^n \,.\, 2^n + (\, 2^n - 1\,)\, 2^n = x \,.\, x + (x - 1\,)\, x = 2\, x^2 - x$$

Por lo tanto, es del orden $O(x^2)$. ■

3.4 El Teorema de no clonación

Como se mencionó en la sección anterior, no es posible copiar o clonar un qubit. Esta afirmación, aparentemente sin demasiada relevancia, tiene importantes consecuencias, como se verá más adelante. Primeramente se demostrará el Teorema:

> ***Teorema 3.1: Teorema de no Clonación***
>
> *No existe ninguna compuerta cuántica que permita copiar o clonar un qubit.*

Demostración

Se demostrará esta afirmación por el absurdo. Es decir, se asumirá que sí existe una compuerta cuántica que puede clonar cualquier qubit. Dicha compuerta debe tener al menos dos qubits de salida: la del qubit a copiar y su propia copia. En consecuencia, como el número de entradas es igual al número de salidas, tendrá al menos dos qubits de entrada. A fin de simplificar la demostración, y sin pérdida de generalidad, se supondrá que la compuerta tiene dos qubits de entrada y por lo tanto, dos qubits de salida.
Sean q_1 y q_2 los qubits de entrada, donde q_1 es el qubit a copiar. Luego, si U es la transformación unitaria asociada a dicha compuerta, debe verificar que

$$U\,(\,|\,q_1\,> \otimes\, |\,q_2\,>\,) = |\;q_1\,> \otimes |\;q_1\,>$$

Es decir,

$$U\,(\,|\;q_1\;\;q_2\;>\,) = |\;q_1\;q_1\,> \qquad [\,1\,]$$

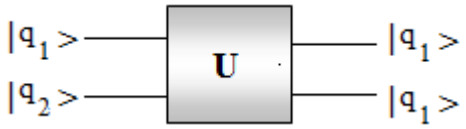


Figura 3.19: Compuerta clonando al qubit q_1.

Sea

$$\{\,|\,0\,>\,,\;|\,1\,>\,\}$$

Una base para el espacio de los qubits**.** Entonces

$$q_1 = \sum_{i=0}^{1} \alpha_i \, | i >$$

Donde

$$\alpha_i \in \mathbb{C} \;\; \forall \; i = 0 , 1 \quad y \quad \sum_{i=0}^{1} | \alpha_i |^2 = 1$$

Si el procedimiento de clonado ha de funcionar, el operador U debe ser capaz de duplicar los vectores de la base, es decir, si $i = 0 , 1$ se verifica:

$$U (| i \;\; q_2 >) = U (| i > \otimes | q_2 >) = | i > \otimes | i > = | i \; i >$$

Luego, al aplicar la propiedad de linealidad de U resulta

$$U (| \; q_1 \;\; q_2 >) = U (| \; q_1 \;\otimes\; q_2 >) \;=\; U \left(\sum_{i=0}^{1} \alpha_i \, | i > \otimes \, q_2 \right) =$$

$$= \sum_{i=0}^{1} \alpha_i \; U \left(| i > \otimes \, q_2 \right) \; = \sum_{i=0}^{1} \alpha_i \;\; | i \;\; i >$$

Por lo tanto,

$$U (| \; q_1 \;\; q_2 >) = \sum_{i=0}^{1} \alpha_i \;\; | i \;\; i > \qquad [2]$$

Por otro lado,

$$| \; q_1 \; q_1 > = | \; q_1 \otimes \; q_1 > = \left(\sum_{i=0}^{1} \alpha_i \, | i > \right) \otimes \left(\sum_{j=0}^{1} \alpha_j \, | j > \right) = \sum_{i=0}^{1} \; \sum_{j=0}^{1} \alpha_i \; \alpha_j \, | i \;\; j >$$

En consecuencia,

$$| \; q_1 \; q_1 > \; = \sum_{i=0}^{1} \; \sum_{j=0}^{1} \alpha_i \; \alpha_j \, | i \;\; j > \qquad [3]$$

Reemplazando [2] y [3] en [1] se obtiene

$$\sum_{i=0}^{1} \alpha_i \;\; | i \;\; i > \; = \sum_{i=0}^{1} \; \sum_{j=0}^{1} \alpha_i \; \alpha_j \, | i \;\; j >$$

Teniendo en cuenta que $| i \; j >$ son linealmente independientes $\forall \; i , j = 0 , 1$ resulta

$$\alpha_i \; \alpha_j = \begin{cases} \alpha_i & \text{si } j = i \\ 0 & \text{si } j \neq i \end{cases}$$

De este modo se llega a un absurdo, ya que esta igualdad no siempre es válida para cualquier qubit q_1. ■

Este Teorema fue formulado por primera vez por William Wootters y Wojciech Zurek en 1982 y sorprendió a toda la comunidad científica, especialmente por sus importantes implicaciones. Algunas de ellas se exponen a continuación.

Consecuencias del Teorema de no Clonación

■ ***En Computación Cuántica***

El Teorema impone fuertes restricciones en esta área, ya que impide realizar copias de seguridad de los datos y almacenarlos en forma permanente. El mismo procedimiento de computación cuántica queda dificultado, ya que no es posible registrar los estados intermedios para la detección y corrección de errores en el proceso.

■ ***En Criptografía Cuántica***

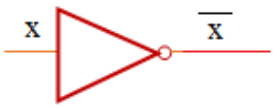
En contra partida, el teorema es esencial para la criptografía cuántica, ya que asegura que es imposible que un tercer observador (espía) copie la señal que las dos partes amigas se intercambian entre sí. Así, si el emisor del mensaje cifrado asegura que cada qubit se envía tan sólo una vez (una sola copia del sistema cuántico que transporta el mensaje), la criptografía cuántica es infalible.

3.5 ¿Puede una compuerta lógica transformarse en una compuerta cuántica?

Como se sabe, las compuertas lógicas convencionales son NOT, AND, OR y XOR. Su tabla de valores y representación gráfica se muestran a continuación:

Compuerta NOT

x	$\bar{x}$
0	1
1	0

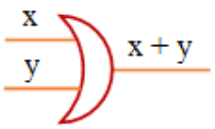


Compuerta AND

x	y	x ∧ y (ó x y)
0	0	0
0	1	0
1	0	0
1	1	1

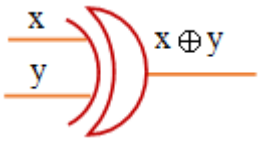
Compuerta OR

x	y	x ∨ y (ó x + y)
0	0	0
0	1	1
1	0	1
1	1	1



Compuerta XOR (ó excluyente o suma módulo 2)

x	y	$x \oplus y$
0	0	0
0	1	1
1	0	1
1	1	0



Las compuertas AND y NOT son llamadas universales, ya que cualquier circuito lógico puede implementarse utilizando sólo estas compuertas.
Pero, ¿podrán estas compuertas lógicas transformarse en cuánticas? Esta pregunta desveló durante mucho tiempo a los especialistas, pues en caso de que se pudiera, todos los algoritmos diseñados para una computadora clásica podrían ser aplicados a un ordenador cuántico. De este modo, cualquier ordenador cuántico podría hacer lo mismo que uno clásico, pero a mayor velocidad.
Como se detalló en la sección anterior, las compuertas cuánticas son reversibles y tienen mismo número de entradas que de salidas. Desafortunadamente, la única compuerta convencional que verifica estas condiciones es la compuerta NOT.

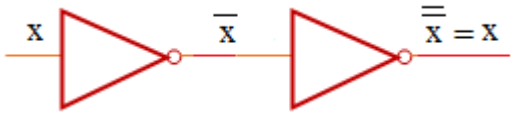


Figura 3.20: La compuerta NOT, única compuerta convencional que es reversible.

Este hecho desilusionó a los científicos, llevándolos a concluir que no es posible convertir una compuerta lógica y en consecuencia un circuito lógico, en un circuito cuántico. Sin embargo, entre 1980 y 1982, Tommas Toffoli y Edward Fredkin, miembros del MIT (Massachusetts Institute of Technology) diseñaron compuertas lógicas reversibles de tres bits de entrada, capaces de ejecutar las operaciones lógicas NOT y AND. En la siguiente sección se las describirá detalladamente.

3.6 La compuerta de Toffoli

Esta compuerta consta de tres bits de entrada y tres bits de salida. Su forma de operar se ilustra en la siguiente figura:

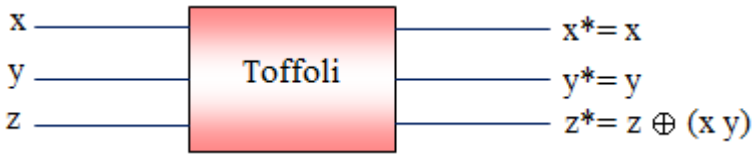


Figura 3.21: La compuerta de Toffoli.

Y su tabla de verdad es

Entrada			*Salida*		
x	**y**	**z**	**x* = x**	**y* = y**	**z* = z ⊕ (x ∧ y)**
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	1	1	1
1	1	1	1	1	0

Se puede comprobar que el último bit z es negado si y sólo si los primeros dos bits x e y valen 1. La compuerta de Toffoli es reversible, ya que al hacerla actuar dos veces se obtiene la identidad.

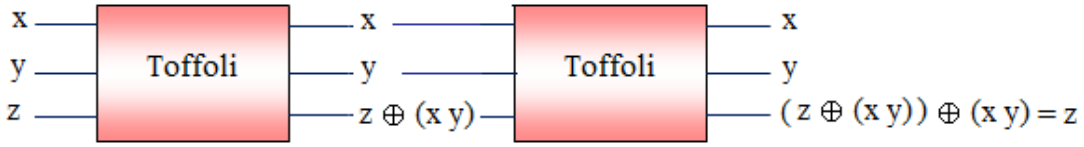


Figura 3.22: Reversibilidad de la compuerta de Toffoli.

En la siguiente figura se muestra el circuito lógico asociado a dicha compuerta:

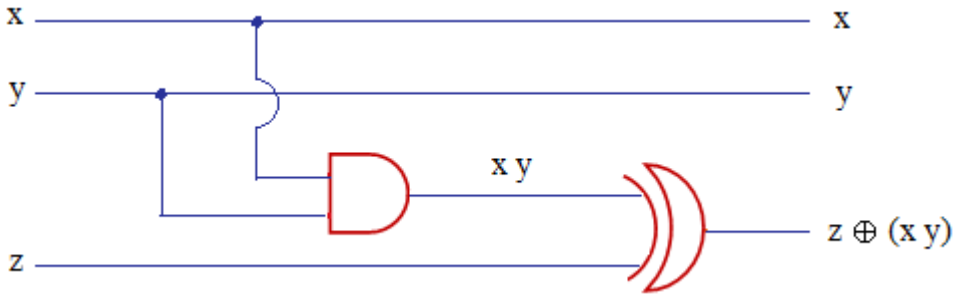


Figura 3.23: Circuito lógico de la compuerta de Toffoli.

Su carácter de compuerta universal se resume en la siguiente ecuación, en donde se indica que puede actuar como una compuerta AND o una compuerta NOT o una compuerta XOR:

$$z^* = z \oplus (x \wedge y) = \begin{cases} x \wedge y & \text{si } z = 0 \quad \text{(compuerta AND)} \\ x \oplus z & \text{si } y = 1 \quad \text{(compuerta XOR)} \\ \bar{z} & \text{si } x = y = 1 \quad \text{(compuerta NOT)} \end{cases}$$

La compuerta de Toffoli puede ser vista como una compuerta cuántica de 3-qubits, si se la define por

$$\mathbf{U}_{\mathbf{Toffoli}} \mid x,\ y,\ z > = \mid x, y,\ z \oplus (x \wedge y) >$$

Para hallar la matriz unitaria asociada a esta compuerta, se considera la siguiente base ortonormal para el sistema formado por 3-qubits:

$$\{ \mid 000 >, \mid 001 >, \mid 010 >, \mid 011 >, \mid 100 >, \mid 101 >, \mid 110 >, \mid 111 > \}$$

Y se calcula

$\mathbf{U_{Toffoli}} \mid 000 > = \mid 0 , 0 , 0 \oplus (0 \wedge 0) > = \mid 000 >$

$\mathbf{U_{Toffoli}} \mid 001 > = \mid 0 , 0 , 1 \oplus (0 \wedge 0) > = \mid 001 >$

$\mathbf{U_{Toffoli}} \mid 010 > = \mid 0 , 1 , 0 \oplus (0 \wedge 1) > = \mid 010 >$

$\mathbf{U_{Toffoli}} \mid 011 > = \mid 0 , 1 , 1 \oplus (0 \wedge 1) > = \mid 011 >$

$\mathbf{U_{Toffoli}} \mid 100 > = \mid 1 , 0 , 0 \oplus (1 \wedge 0) > = \mid 100 >$

$\mathbf{U_{Toffoli}} \mid 101 > = \mid 1 , 0 , 1 \oplus (1 \wedge 0) > = \mid 101 >$

$\mathbf{U_{Toffoli}} \mid 110 > = \mid 1 , 1 , 0 \oplus (1 \wedge 1) > = \mid 111 >$

$\mathbf{U_{Toffoli}} \mid 111 > = \mid 1 , 1 , 1 \oplus (1 \wedge 1) > = \mid 110 >$

Luego, la matriz correspondiente a esta compuerta es

$$U_{\text{Toffoli}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Que claramente resulta unitaria.

3.7 La compuerta de Fredkin

Esta compuerta también consta de tres bits de entrada y tres bits de salida. En este caso, los dos últimos bits y, z se intercambian si el primer bit x es 0. En caso contrario, permanecen iguales. La tabla de valores de esta compuerta se muestra a continuación:

Entrada			***Salida***		
x	**y**	**z**	**x*= x**	**y***	**z***
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	1
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	1	1	0
1	1	1	1	1	1

Para construir la función booleana correspondiente a la salida y* en términos de las entradas x, y, z, se suman los mintérminos. Éstos se obtienen multiplicando las variables x, y, z, negando aquéllas que valen 0, pero considerando sólo los casos en que y* sea

igual a 1. Luego se aplican las propiedades aritméticas del Álgebra de Boole. De este modo se obtiene

$$y^* = \bar{x}\ \bar{y}\ z + \bar{x}\ y\ z + x\ y\ \bar{z} + x\ y\ z = \bar{x}\ z\ (\bar{y} + y) + x\ y\ (\bar{z} + z) = \bar{x}\ z\ (1) + x\ y\ (1) =$$
$$= \bar{x}\ z + x\ y$$

De manera similar resulta

$$z^* = \bar{x}\ y\ \bar{z} + \bar{x}\ y\ z + x\ \bar{y}\ z + x\ y\ z = \bar{x}\ y\ (\bar{z} + z) + x\ z\ (\bar{y} + y) = \bar{x}\ y\ (1) + x\ z\ (1) =$$
$$= \bar{x}\ y + x\ z$$

Por lo tanto, la forma de operar esta compuerta en términos de las compuertas lógicas convencionales es la siguiente:

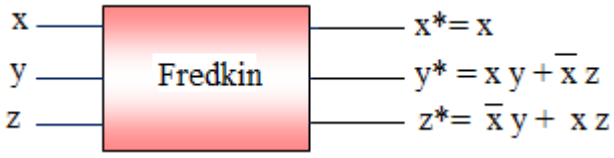


Figura 3.24: La compuerta de Fredkin.

Se puede verificar que la compuerta de Fredkin es también reversible, ya que al aplicarla dos veces se obtiene la identidad. Su carácter de compuerta universal se resume en la siguiente ecuación, la cual indica que puede actuar como una compuerta AND o una compuerta NOT:

$$y^* = \bar{x}\ z + x\ y\ = \begin{cases} x \wedge y & \text{si } z = 0 \quad \text{(compuerta AND)} \\ \bar{x} & \text{si } y = 0\ ,\ z = 1 \quad \text{(compuerta NOT)} \end{cases}$$

La compuerta de Fredkin también puede ser vista como una compuerta cuántica de 3-qubits, si se la define por

$$\mathbf{U_{Fredkin}}\ |\ x,\ y\ ,\ z\ > = |\ x\ ,\ x\ y + \bar{x}\ z,\ \bar{x}\ y + x\ z\ >$$

De manera similar a la probada para la compuerta de Toffoli, se puede demostrar que la matriz asociada a la compuerta de Fredkin es

$$U_{Fredkin} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Que es una matriz unitaria.
Las compuertas de Toffoli y de Fredkin revolucionaron la Computación Cuántica ya que se logró demostrar, al menos teóricamente, que todos los circuitos lógicos pueden implementarse en un circuito cuántico.

En Marzo de 2016, investigadores de la Universidad de Giffith y de Queensland (Australia) anunciaron que construyeron una compuerta cuántica de Fredkin, capaz de intercambiar qubits. Este hecho pone de manifiesto que la Computación Cuántica es una ciencia que recién comienza y que la materialización de sus resultados teóricos se está llevando a cabo desde hace muy poco tiempo.

Para finalizar este capítulo y comprender mejor el funcionamiento de las compuertas cuánticas, se mostrará un circuito cuántico capaz de transmitir información hacia un lugar arbitrariamente alejado. Este hecho se conoce como ***teleportación cuántica***, concepto que se desarrollará en la próxima sección.

3.8 La Teleportación Cuántica

Esta técnica, sorprendente y no trivial, fue descubierta de forma teórica en 1993 por varios científicos, entre ellos, Charles Bennet (IBM Research Division, New York) y Richard Josza (Déparetment IRO, Université de Montréal) y concretada por primera vez de manera experimental en 1997, utilizando medios ópticos. Su definición es la siguiente:

Definición 3.3: Teleportación Cuántica

La Teleportación Cuántica es un proceso que consiste en transmitir un estado cuántico desde un lugar A a otro lugar B arbitrariamente alejado de A y sin ningún medio físico que los una, por medio de un estado cuántico entrelazado que es compartido entre A y B.

La teleportación cuántica no involucra transporte de materia o energía, sino sólo de información. Este hecho rompe el mito de creer que la teleportación cuántica es capaz de transferir materia hacia cualquier lugar del espacio, como ocurre en las series de ciencia ficción.

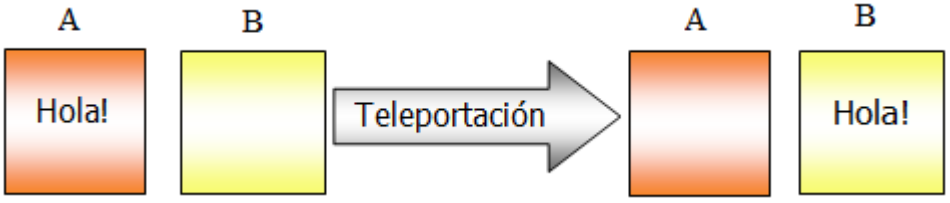


Figura 3.25: Ejemplo comparativo de teleportación cuántica: Se puede transferir la información contenida en la hoja situada en la posición A hacia la hoja situada en B. Pero la hoja situada en A no se puede transportar a B.

Tampoco trae aparejada ninguna contradicción con la Teoría de la Relatividad (es decir, no implica transmisión de información a velocidades mayores que la velocidad de la luz), pues A y B deben intercambiar cierta información por medios convencionales. Un aspecto notable es que el estado a teletransportar no necesita ser conocido.
Debido a que la teleportación utiliza un estado cuántico entrelazado, se verá primeramente cómo generar este tipo de estados desde el punto de vista matemático.

Cómo generar un sistema de dos qubits entrelazados

Sean $q_1 = |x>$ y $q_2 = |y>$ dos qubits con x, y $\in \{0,1\}$. Entonces, se puede generar entre ellos un estado entrelazado mediante el siguiente circuito cuántico:

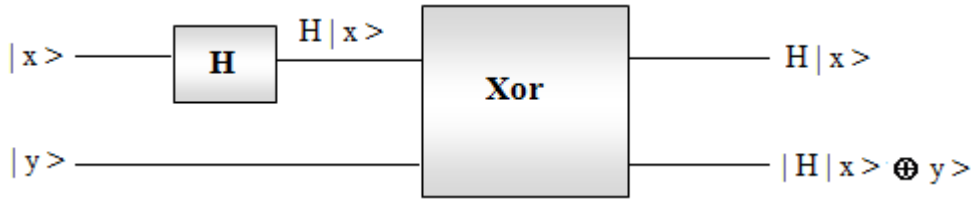


Figura 3.26: Circuito cuántico que permite generar dos qubits q_1 y q_2 entrelazados, y que originalmente estaban en estado x e y respectivamente.

Este circuito puede representarse también, de manera más simplificada, del siguiente modo:

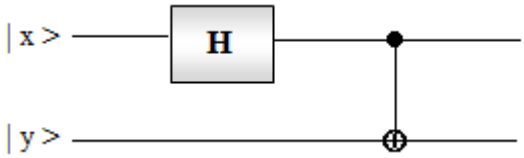


Figura 3.27: Circuito cuántico que permite generar dos qubits entrelazados, expresado en forma simplificada.

Según el Cuarto Postulado de la Mecánica Cuántica, el estado inicial del sistema formado por los qubits $|x>$ e $|y>$ queda determinado por el producto tensorial

$$|x> \otimes |y>$$

Luego de atravesar las compuertas H y Xor el sistema adquiere, de acuerdo a los valores de los qubits $|x>$ e $|y>$, uno de los siguientes estados entrelazados:

$$\beta_{xy} = U_{xor}(H|x> \otimes |y>) \quad \text{con } x, y \in \{0,1\}.$$

Éstos reciben el nombre de "Estados de Bell". Por ejemplo, si x = y = 0 entonces

$$\beta_{00} = U_{xor}(H|0> \otimes |0>) = U_{xor}\left(\frac{1}{\sqrt{2}}(|0> + |1>) \otimes |0>\right) =$$

$$= U_{xor}\left(\frac{1}{\sqrt{2}}(|0> \otimes |0> + |1> \otimes |0>)\right) = U_{xor}\left(\frac{1}{\sqrt{2}}(|00> + |10>)\right) =$$

$$= U_{xor}\left(\frac{1}{\sqrt{2}}(|00> + |10>)\right) = \frac{1}{\sqrt{2}}(U_{xor}|00> + U_{xor}|10>) =$$

$$= \frac{1}{\sqrt{2}}(|00> + |11>)$$

Los cálculos para los demás valores de x e y se efectúan de manera similar. La siguiente tabla ilustra los estados entrelazados que pueden obtenerse mediante este circuito:

Entrada		*Salida*
| x >	**| y >**	
| 0 >	| 0 >	$\beta_{00} = \frac{1}{\sqrt{2}}(\mid 00 > + \mid 11 >)$
| 0 >	| 1 >	$\beta_{01} = \frac{1}{\sqrt{2}}(\mid 01 > + \mid 10 >)$
| 1 >	| 0 >	$\beta_{10} = \frac{1}{\sqrt{2}}(\mid 00 > - \mid 11 >)$
| 1 >	| 1 >	$\beta_{11} = \frac{1}{\sqrt{2}}(\mid 01 > - \mid 10 >)$

Por ejemplo, si dos qubits están entrelazados y su estado se describe mediante el estado de Bell $\beta_{00} = \frac{1}{\sqrt{2}}(\mid 00 > + \mid 11 >)$ significa que, en el momento de observar o medir los dos qubits que componen el sistema, ambos estarán en estado 0 o bien ambos estarán en estado 1, ya que la probabilidad de que ambos estén en el mismo estado es igual a 1 (confrontar con la Definición 2.6).

Antes de formalizar matemáticamente el proceso de teleportación de un qubit, primeramente se explicará su procedimiento en forma coloquial, a fin de comprender mejor su funcionamiento.

Cómo efectuar la teleportación cuántica de un qubit.

Se tienen tres qubits: q , q_1 y q_2. Los qubits q y q_1 están con Alice, mientras que el qubit q_2 está con Bob, quien está a millones de kilómetros de distancia de Alice.
Alice (emisor) desea transmitir a Bob (receptor) la información acerca del estado del qubit q que ella misma desconoce. Para llevar a cabo la teleportación cuántica, se genera un estado entrelazado entre los qubits q_1 y q_2.

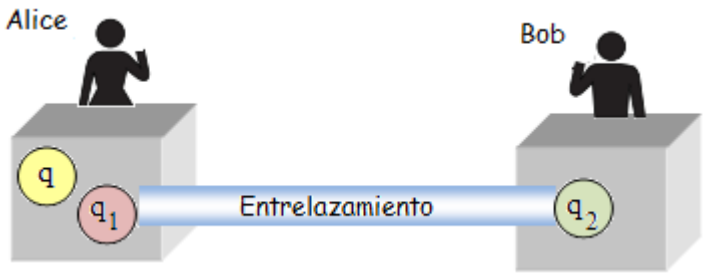


Figura 3.28:Alice y Bob separados a gran distancia y sin ningún medio físico que los una.

Alice diseña un circuito cuántico de dos entradas: la del qubit q, cuyo estado quiere transferir, y la del qubit q_1, que está entrelazado con q_2. Finalmente, ella decide medir los estados de los qubits q y q_1. Cuando esto ocurre, el entrelazamiento entre q_1 y q_2 desaparece.
Mientras tanto, el qubit q_2 sigue estando en un estado desconocido. Alice toma el teléfono e informa a Bob qué mediciones obtuvo de sus dos qubits, las cuales pudieron haber sido 00, 01, 10 ó 11. De este modo Alice, mediante un medio físico convencional, comunica a Bob el valor de dos bits clásicos.

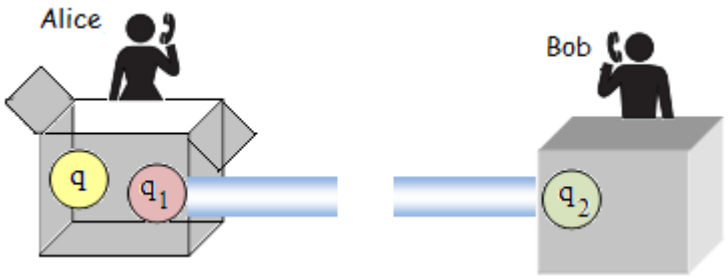


Figura 3.29: Alice comunica a Bob el estado de los qubits q y q_1, una vez hecha la medición.

En base al par de bits enviados por Alice, Bob transforma el qubit q_2 mediante una compuerta cuántica y así recupera en q_2 la información que originalmente estaba en q.

Explicación matemática de cómo se realiza la teleportación cuántica de un qubit

Los pasos a seguir por Alice y Bob son los siguientes:

■ Alice y Bob preparan un estado entrelazado entre dos qubits q_1 y q_2. Por ejemplo, se supondrá que crearon el estado entrelazado $\beta_{00} = \frac{1}{\sqrt{2}}(\mid 00 > + \mid 11 >)$.

■ Alice y Bob se separan. Alice se queda con q_1, el primer qubit del par entrelazado y Bob con q_2, el segundo par entrelazado.

■ Alice quiere ahora transmitir a Bob el estado del qubit $q = \alpha \mid 0 > + \beta \mid 1 >$ que es para ella desconocido.
De este modo se genera un sistema formado por tres qubits: q, q_1 y q_2, con lo cual el estado inicial de dicho sistema quedará descripto mediante un 3-qubit, que se llamará **w**.
■ Alice aplica la compuerta cuántica Xor a sus dos qubits: q y q_1.
■ Alice aplica luego la compuerta cuántica de Hadamard al qubit q.
■ Alice realiza una medición sobre ambos qubits y obtiene dos bits clásicos b_1 y b_2, que envía a Bob por un canal de comunicación convencional.

■ Bob aplica la transformación $Z^{b1}\ X^{b2}$ sobre su qubit, de acuerdo a los bits recibidos b_1, b_2 y donde X , Z son las matrices de Pauli:

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \ , \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

El resultado obtenido por Bob será q.
El esquema completo de la teleportación cuántica se muestra en la figura siguiente:

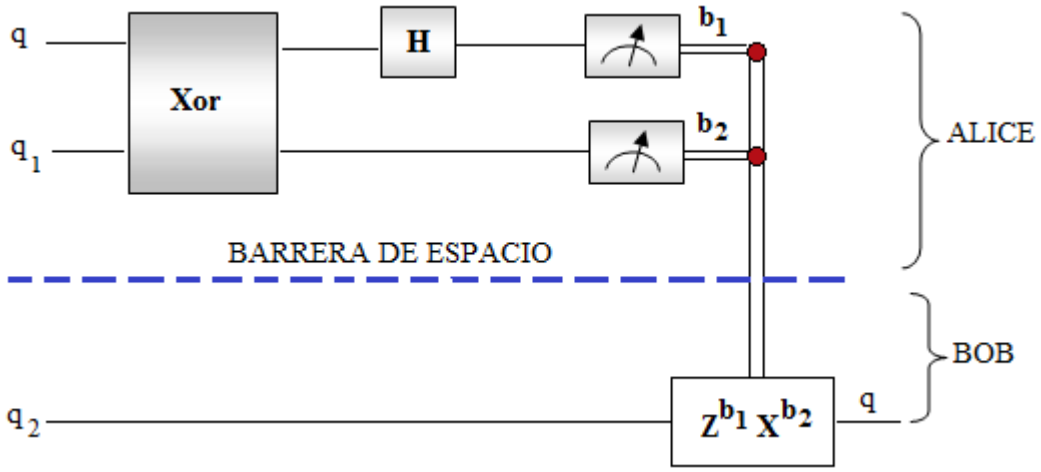


Figura 3.30: Esquema completo de la teleportación cuántica.

Se pueden seguir los estados sucesivos del sistema aplicando paso a paso los operadores correspondientes. Los detalles se dan a continuación:

Estado del sistema antes de ingresar a la compuerta Xor

Según el Cuarto Postulado de la Física Cuántica, el estado del sistema formado por los tres qubits q, q_1 y q_2 queda determinado por el producto tensorial

$$q \otimes \beta_{00} =$$

$$= (\alpha\,|0> + \beta\,|1>) \otimes \frac{1}{\sqrt{2}}(|00> + |11>) = \frac{1}{\sqrt{2}}\left[\alpha\,|0>(|00> + |11>) + \beta\,|1>(|00> + |11>)\right]$$

$$= \frac{1}{\sqrt{2}}\left[\alpha\,(|000> + |011>) + \beta\,(|100> + |111>)\right]$$

De este modo, el estado inicial del sistema queda descripto mediante el 3-qubit

$$\mathbf{w} = \frac{1}{\sqrt{2}}\left[\alpha\,(|000> + |011>) + \beta\,(|100> + |111>)\right]$$

Estado del sistema luego de aplicar la compuerta Xor

Se debe tener en cuenta que la transformación unitaria asociada a la compuerta Xor opera sobre un 2-qubit de la siguiente manera

$$Uxor\,|\,x\ y> = |\,x\ ,\ x \oplus y> \quad \text{para } x,\ y \in \{\,0\ ,\ 1\,\}$$

En este caso, la compuerta se aplica a los qubits q y q_1, cuyos estados están determinados, respectivamente, por las componentes x e y de cada 3-qubit $|\,x\ y\ z>$ de **w**. Por lo tanto, el proceso de atravesar la compuerta Xor queda determinado mediante una transformación unitaria U_1 definida del siguiente modo:

$$U_1\,|\,x\ y\ z> = Uxor\,|\,x\ y>\ |\,z\ > = |\,x\ ,\ x \oplus y>\ |\,z\ >$$

Luego, teniendo en cuenta la propiedad de linealidad de U_1, el sistema evoluciona hacia el siguiente estado:

$$\mathbf{w_1} = U_1\, w = U_1\left(\frac{1}{\sqrt{2}}\left[\alpha\ \left(|000> + |011>\right)+\beta\ \left(|100> + |111>\right)\right]\right) =$$

$$= \frac{1}{\sqrt{2}}\left[\alpha\ \left(U_1|000> + U_1|011>\right)+\beta\ \left(U_1|100> + U_1|111>\right)\right] =$$

$$= \frac{1}{\sqrt{2}}\left[\alpha\ \left(U_{xor}|00>|0> + U_{xor}|01>|1>\right)+\beta\ \left(U_{xor}|10>|0> + U_{xor}|11>|1>\right)\right] =$$

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

$$= \frac{1}{\sqrt{2}}\left[\alpha\ \left(|000> + |011>\right)+\beta\ \left(|11\,0> + |10\,1>\right)\right]$$

Es decir,

$$\mathbf{w_1} = \frac{1}{\sqrt{2}}\left[\alpha\ \left(|000> + |011>\right)+\beta\ \left(|11\,0> + |10\,1>\right)\right]$$

Estado del sistema luego de aplicar la compuerta H

Esta compuerta opera sobre el qubit q, cuyo estado se describe por la componente x de cada 3-qubit | x y z > de **w1.** Por lo tanto, el proceso de atravesar la compuerta H queda determinado mediante la siguiente transformación unitaria U_2:

$$U_2\,|\,x\ y\ z> = H\,|\,x\ >\ |\,y\ z\ >$$

Luego, aplicando nuevamente la propiedad de linealidad de U_2 , el sistema evoluciona hacia el siguiente estado:

$$\mathbf{w_2} = U_2\, w_1 = U_2\left(\frac{1}{\sqrt{2}}\left[\alpha\ \left(|000> + |011>\right)+\beta\ \left(|11\,0> + |10\,1>\right)\right]\right) =$$

$$= \frac{1}{\sqrt{2}}\left[\alpha\ \left(U_2\ |000> + U_2\ |011>\right)+\beta\ \left(U_2\ |11\,0> + U_2\,|10\,1>\right)\right] =$$

$$= \frac{1}{\sqrt{2}}\left[\alpha\ \left(H\,|0>\ |00> + H\,|0>\ |11>\right)+\beta\ \left(H\,|1>\ |1\,0> + H\,|1>\,|0\,1>\right)\right]$$

Teniendo en cuenta que

$$H\,|\,0>\ =\ \frac{1}{\sqrt{2}}\ |\,0> + \frac{1}{\sqrt{2}}\ |\,1>$$

$$H\,|\,1> = \frac{1}{\sqrt{2}}\ |\,0> - \frac{1}{\sqrt{2}}\ |\,1>$$

Resulta

$$\mathbf{w_2} = \frac{1}{2}\ [\alpha\ \left((|0> + |1>)\,|00> + (|0> + |1>)\,|11>\right)\ +$$

$$+\ \beta\ \left((|0> - |1>)\,|1\,0> + (|0> - |1>)\,|0\,1>\right)\ ] =$$

$$\frac{1}{2}\left[\alpha\ \left(|000>+|100>+\ |011>+|111>\right)+\beta\ \left(|010>-|110>+|001>-|101>\right)\right]=$$

$$=\frac{1}{2}\ \left[\ |00>\left(\alpha|0>+\beta|1>\right)+\ |01>\left(\alpha|1>+\beta|0>\right)+\ |10>\left(\alpha|0>-\beta|1>\right)+\right.$$

$$\left.+\ |11>\left(\alpha|1>-\beta|0>\right)\right]=\frac{1}{2}\sum_{b_1,b_2=\ 0}^{1}\ |\ b_1b_2>\left(X^{b_2}\ Z^{b_1}\right)q$$

En consecuencia, el sistema queda en el siguiente estado:

$$\mathbf{w_2}=\frac{1}{2}\sum_{b_1,b_2=\ 0}^{1}\ |\ b_1\ b_2>\left(X^{b_2}\ Z^{b_1}\right)q$$

Donde b_1 indica el estado del qubit q y b_2 el estado del qubit q_1.

Alice mide sus dos qubits

Los qubits que observa Alice son q y q_1, que corresponden a los dos primeros qubits del sistema. Al medirlos, obtiene uno de los cuatro valores b_1 b_2 posibles, con $b_1, b_2 \in \{\ 0\ ,1\ \}$. Por la Proposición 2.4, el sistema colapsa al estado

$$\mathbf{w_3}=|\ b_1\ b_2>\left(X^{b_2}\ Z^{b_1}\right)q$$

Es decir,

$$\mathbf{w_3}=|\ b_1\ b_2\ \ \left(X^{b_2}\ Z^{b_1}\right)q>$$

Así, el tercer qubit del sistema que es q_2 y lo tiene Bob, quedó en el siguiente estado:

$$\mathbf{q_2}=\left(X^{b_2}\ Z^{b_1}\right)q$$

Pues

■ Si $b_1 = 0$, $b_2 = 0$ entonces

$$\left(X^{b_2}\ Z^{b_1}\right)q=q=\alpha\ |\ 0>+\beta\ |\ 1>$$

Luego

$$w_3=|\ b_1\ b_2\ \ \left(X^{b_2}\ Z^{b_1}\right)q>=\ =|\ 0\ 0\ \left(\alpha|0>+\beta|1>\right)>=\alpha\ |\ 0\ 0\ 0>+\beta\ |\ 0\ 0\ 1>$$

Por otro lado, si se define

P ($X_3 = x$) : La probabilidad de encontrar el tercer qubit del sistema (q_2) en estado x
(con x = 0 ó x = 1)

Resulta

$$P(X_3=0)=|\ \alpha\ |^2$$

$$P(X_3=1)=|\ \beta\ |^2$$

En consecuencia, q_2 tiene el mismo estado que $\left(X^{b_2}\ Z^{b_1}\right) q$. Por lo tanto,

$$q_2 = \left(X^{b_2}\ Z^{b_1}\right) q$$

■ Si $b_1 = 0$, $b_2 = 1$ entonces

$\left(X^{b_2}\ Z^{b_1}\right) q = \beta \mid 0 > + \alpha \mid 1 >$

Luego

$w_3 = \mid b_1\ b_2\ \left(X^{b_2}\ Z^{b_1}\right) q > = \mid 0\ 0\ (\beta \mid 0 > + \alpha \mid 1>) > = \beta \mid 0\ 0\ 0 > + \alpha \mid 0\ 0\ 1 >$

Por lo tanto

$P(X_3 = 0) = \mid \beta \mid^2$

$P(X_3 = 1) = \mid \alpha \mid^2$

De aquí resulta que q_2 tiene el mismo estado que $\left(X^{b_2}\ Z^{b_1}\right) q$. En consecuencia,

$$q_2 = \left(X^{b_2}\ Z^{b_1}\right) q$$

De manera similar se demuestra para los demás valores de b_1 y b_2.

Alice envía la información a Bob

Una vez que Alice envió a Bob los valores de b_1 y b_2, Bob aplicará a q_2 la compuerta $Z^{b_1}\ X^{b_2}$. Entonces

$$Z^{b_1}\ X^{b_2}\ q_2 = Z^{b_1}\ X^{b_2} \left(X^{b_2} Z^{b_1} q\right) = Z^{b_1} \left(X^{b_2} X^{b_2}\right) Z^{b_1} q = Z^{b_1} I\ Z^{b_1} q = Z^{b_1} Z^{b_1} q = Iq = q$$

Donde I es la matriz Identidad.
De este modo, Bob recuperó en q_2 el estado del qubit q. ■

Debe recalcarse que, para que se produzca la teleportación cuántica, es necesario que Alice le comunique a Bob el resultado de sus mediciones mediante un medio de comunicación convencional. Este hecho impide que haya comunicación supralumínica entre Alice y Bob, con lo cual no se está violando la Teoría de la Relatividad, que afirma que ningún objeto puede superar la velocidad de la luz.
Otra curiosidad de la teleportación es que "parecería crear una copia del qubit original" (con lo cual se estaría violando el Teorema de no Clonación). Pero esto no es así ya que, durante el proceso, el qubit original termina degradado en uno de los dos estados finales $\mid 0 >$ ó $\mid 1 >$.
Finalmente, cabe mencionar que en Junio de 2017 un grupo de científicos de la Universidad de Ciencia y Tecnología de China, en colaboración con la Academia de Ciencias de Austria, lograron teletransportar el estado de un fotón a una distancia de 1203 km desde un satélite hasta una estación en la Tierra, pudiendo de este modo romper marcas anteriores.
Con esta tecnología en plena investigación, los ingenieros se afanan en desarrollar una nueva generación de ordenadores cuánticos ultrapotentes cuyos sistemas no sólo serán mucho más veloces, también serán imposibles de hackear; si un pirata informático intercepta uno de los fotones entrelazados, el otro lo sabrá. Sin embargo, aún se está en los albores de entender completamente el fenómeno de teleportación.

Capítulo 4

El paralelismo cuántico

Capítulo 4: El paralelismo cuántico

El paralelismo cuántico es la propiedad que tienen los algoritmos cuánticos de efectuar varias operaciones al mismo tiempo, en virtud del estado de superposición que pueden adquirir las partículas subatómicas.
A diferencia de los algoritmos clásicos, el paralelismo cuántico es el principal responsable del enorme potencial de cómputo que gozan los algoritmos cuánticos. Gracias a esta propiedad, las computadoras cuánticas pueden valuar, en forma simultánea, una función f(x) para múltiples valores de x.
El Algoritmo de Deutsch- Josza, uno de los primeros algoritmos cuánticos, es un claro ejemplo que permite apreciar el funcionamiento y la potencia de los programas cuánticos.
Propuesto por David Deutsch y Richard Josza en 1992, su finalidad es determinar si una función booleana $f(x_1, x_2, \ldots, x_n)$ de n variables es constante (vale 0 en todas las entradas o 1 en todas las entradas) o está balanceada (es decir, si toma el valor 1 para la mitad de las entradas y 0 para la otra mitad).
En este capítulo sólo se explicará el algoritmo para funciones booleanas de una variable, ya que simplemente se pretende mostrar cómo se logra el paralelismo cuántico mediante la generación de estados de superposición cuántica.

4.1 Nociones previas

Según se detalló en el Capítulo 3, la compuerta de Hadamard opera sobre un qubit del modo siguiente:

$$H \mid 0 > = \frac{1}{\sqrt{2}} \mid 0 > + \frac{1}{\sqrt{2}} \mid 1 >$$

$$H \mid 1 > = \frac{1}{\sqrt{2}} \mid 0 > - \frac{1}{\sqrt{2}} \mid 1 >$$

Así, esta compuerta transforma los qubits $\mid 0 >$ y $\mid 1 >$ en estados de superposición, permitiendo estar en estado 0 y 1 al mismo tiempo.
Por otro lado, dada una función booleana de una variable $f: \{ 0 , 1 \} \rightarrow \{ 0 , 1 \}$, la compuerta cuántica **f**, que opera sobre dos qubits, se define de la siguiente manera:

$$U_f \mid x \;\; y > = \mid x , \; y \oplus f(x) >$$

Donde $\oplus$ representa la suma módulo 2 y $x , y \in \{ 0 , 1 \}$.Dicha compuerta realiza las siguientes operaciones:

$U_f \mid 0 \; 0 > = \mid 0 , \; 0 \oplus f(0) > = \mid 0 , \; f(0) > = \mid 0 > \mid f(0) >$

$U_f \mid 0 \; 1 > = \mid 0 , \; 1 \oplus f(0) > = \mid 0 , \; \overline{f(0)} > = \mid 0 > \mid \overline{f(0)} >$

$U_f \mid 1 \; 0 > = \mid 1 , \; 0 \oplus f(1) > = \mid 1 , \; f(1) > = \mid 1 > \mid f(1) >$

$U_f \mid 1 \; 1 > = \mid 1 , \; 1 \oplus f(1) > = \mid 1 , \; \overline{f(1)} > = \mid 1 > \mid \overline{f(1)} >$

Donde $\overline{f(x)}$ es el complemento de f(x), para x = 0 ó 1. Es decir,

$$\overline{f(x)} = \begin{cases} 1 & \text{si } f(x) = 0 \\ 0 & \text{si } f(x) = 1 \end{cases}$$

4.2 Algoritmo de Deutsch-Josza para funciones booleanas de una variable

Como se mencionó anteriormente, dada una función booleana f: $\{0, 1\} \rightarrow \{0, 1\}$, el algoritmo cuántico de Deutsch-Josza determinará si f(x) es constante o no.
El circuito cuántico que implementa el algoritmo tiene dos entradas: | 0 > y | 1 >. Se lo muestra a continuación:

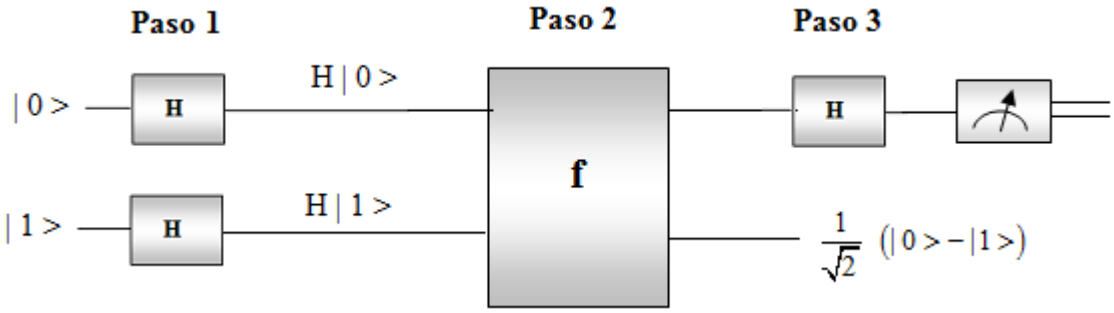


Figura 4.1: Circuito cuántico del algoritmo de Deutsch-Josza

Las operaciones que ejecuta el circuito son las siguientes:

Paso 1

Se aplica la compuerta de Hadamard a los qubits de entrada |0 > y | 1 > :

$$H\,|\,0> \;=\; \frac{1}{\sqrt{2}}\,|\,0> + \frac{1}{\sqrt{2}}\,|\,1>$$

$$H\,|\,1> \;=\; \frac{1}{\sqrt{2}}\,|\,0> - \frac{1}{\sqrt{2}}\,|\,1>$$

Paso 2

Se aplica la compuerta **f** al 2-qubit H | 0 > ⊗ H | 1 >. Previamente se calculará este producto tensorial:

$$H\,|\,0> \otimes H\,|1> \;= \left(\frac{1}{\sqrt{2}}\,|\,0> + \frac{1}{\sqrt{2}}\,|\,1>\right) \otimes \left(\frac{1}{\sqrt{2}}\,|\,0> - \frac{1}{\sqrt{2}}\,|\,1>\right) =$$

$$= \frac{1}{2}\left(\,|\,0\,0> - |\,0\,1> + |\,1\,0> - |\,1\,1>\,\right)$$

Luego, al atravesar la compuerta **f,** y teniendo en cuenta que ésta representa una transformación lineal, resulta:

$$U_f\left(H\,|\,0> \otimes H\,|\,1>\right) = U_f\left(\frac{1}{2}\left(\,|\,0\,0> - |\,0\,1> + |\,1\,0> - |\,1\,1>\,\right)\right) =$$

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

$$= \frac{1}{2}\left(|0> \ |f(0)> - |0> \ |\overline{f(0)}> + |1> \ |f(1)> - |1> \ |\overline{f(1)}> \right) =$$

$$= \frac{1}{2}\left(|0> \ (-1)^{f(0)} \ (|0> - |1>) + |1> \ (-1)^{f(1)} \ (|0> - |1>) \right) =$$

$$= \frac{1}{2}\left(\left(|0> \ (-1)^{f(0)} + |1> \ (-1)^{f(1)} \right) \ (|0> - |1>) \right) =$$

$$= \underbrace{\frac{1}{\sqrt{2}} \left(|0> \ (-1)^{f(0)} + |1> \ (-1)^{f(1)} \right)}_{\text{Primer qubit}} \quad \underbrace{\frac{1}{\sqrt{2}} \ (|0> - |1>)}_{\text{Segundo qubit}}$$

El estado de superposición del primer qubit resultante es notable, pues contiene simultáneamente los valores f(0) y f(1). Es aquí donde se destaca el paralelismo cuántico, ya que se está calculando al mismo tiempo f(0) y f(1), que son todos los valores que toma la función f. Por esta razón se dice que el Algoritmo de Deutsch-Josza es como "mirar a la vez las dos caras de una moneda".
A diferencia de lo que ocurre con los circuitos clásicos, donde el paralelismo se logra mediante circuitos múltiples, y cada uno computando f(x) para distintos valores de x, aquí un solo circuito y una sola evaluación computa ambos valores.

Paso 3

El primer qubit del paso anterior será el que determinará si la función es o no constante. Primeramente se le aplicará una compuerta de Hadamard, para luego someterlo a una medición. Al atravesar esta compuerta se obtiene:

$$H\left(\frac{1}{\sqrt{2}} \left(|0> \ (-1)^{f(0)} + |1> \ (-1)^{f(1)} \right) \right) = \frac{1}{\sqrt{2}} \ H\left(|0> \ (-1)^{f(0)} + |1> \ (-1)^{f(1)} \right) =$$

$$= \frac{1}{\sqrt{2}} \ (-1)^{f(0)} H|0> + \frac{1}{\sqrt{2}} \ (-1)^{f(1)} H|1> \ = \frac{1}{\sqrt{2}} \ (-1)^{f(0)} \left(\frac{|0> + |1>}{\sqrt{2}} \right) + \frac{1}{\sqrt{2}} \ (-1)^{f(1)} \left(\frac{|0> - |1>}{\sqrt{2}} \right) =$$

$$= \frac{1}{2} \ |0> \left((-1)^{f(0)} + (-1)^{f(1)} \right) + \frac{1}{2} \ |1> \left((-1)^{f(0)} - (-1)^{f(1)} \right) = \begin{cases} \pm \ |0> & \text{si } f(0) = f(1) \\ \pm \ |1> & \text{si } f(0) \neq f(1) \end{cases}$$

En consecuencia, si al efectuar la medición se obtiene 0, significará que la función es constante. Mientras que si resulta 1, la función es balanceada. ■

Capítulo 5

Exponenciación modular

Capítulo 5: Exponenciación modular

La exponenciación modular consiste en calcular

$$x^a \pmod n$$

Donde $x, a \in \mathbb{Z} \wedge n \in \mathbb{N}$. Se supone que tanto "x" como "a" y "n" son números de L-bits (es decir, de L dígitos binarios). El valor "L" dependerá de la capacidad de almacenamiento del procesador.

Para el caso del Algoritmo de Shor, "n" será el número a factorizar, "a" representará una superposición de estados, mientras que "x" será un número fijo.

La operación potencia modular es equivalente a calcular matemáticamente x^a, luego dividir el resultado obtenido por "n" y finalmente tomar el resto de dicha división entera. Sin embargo, calcularla de este modo implica varios problemas en cuanto a su implementación práctica. El más grave de ellos es el de almacenamiento. Si "x" y "a" son números grandes (algo común en los sistemas criptográficos) es posible que los requerimientos de almacenamiento del resultado x^a sean imposibles de cumplir. Por esta razón se han propuesto algoritmos para el cálculo más eficiente de la potencia modular.

De hecho, la parte del algoritmo de Shor que consume la mayor parte del tiempo y espacio de memoria es precisamente la exponenciación modular.

En este capítulo se mostrará un algoritmo para calcular la potencia modular, que requerirá O(L) espacio de memoria y $O(L^3)$ tiempo de ejecución para computar $x^a \pmod n$.

Por otro lado, se mostrará cómo se diseña la compuerta cuántica para el cálculo de la exponenciación modular.

5.1 Algoritmo para el cálculo de la exponenciación modular

Para calcular $x^a \pmod n$ se debe considerar, en primer lugar, el número "a" en su expresión binaria. Teniendo en cuenta que "a" debe tener L-dígitos binarios, resulta

$$a = a_{L-1}\, a_{L-2} \ldots a_1\, a_0$$

Con $a_i = 0$ ó 1, $\forall i = 0, \ldots, L-1$

De este modo, "a" expresado en base 10 es

$$a = \sum_{i=0}^{L-1} a_i\, 2^i$$

Por lo tanto

$$x^a = x^{\sum_{i=0}^{L-1} a_i\, 2^i} = \prod_{i=0}^{L-1} x^{a_i\, 2^i} \pmod n$$

Este resultado da origen al algoritmo para calcular $x^a \pmod n$, cuyo pseudocódigo es el siguiente:

```
Potencia : = 1
Para i = 0 hasta L - 1 hacer
    Calcular x^(2^i) (mod n)
    Si a_i = 1 entonces
        Potencia : = Potencia · x^(2^i)
    Fin Si
Fin Para
```

El resultado de $x^a \pmod n$ queda almacenado en la variable Potencia. Para el cálculo de x^{2^i} se utiliza la siguiente propiedad algebraica:

$$x^{2^1} = x^2$$

$$x^{2^2} = \left(x^2\right)^2$$

$$x^{2^3} = \left(\left(x^2\right)^2\right)^2$$

Y en general,

$$x^{2^i} = \underbrace{\left(\left(\left(x^2\right)^2\right)^2 \ldots\ldots\right)^2}_{\text{i-veces}}$$

Luego, el pseudocódigo del algoritmo queda descripto del siguiente modo:

```
Potencia = 1
x : = resto(x, n)
    Si a_0 = 1 entonces
    Potencia : =  x
    Fin Si
Para i = 1 hasta L - 1 hacer
  *  Cálculo de x^(2^i) (mod n) *
    x : = x^2
    x : = resto(x, n)
    Si a_i = 1 entonces
    Potencia : = Potencia · x
    Fin Si
Fin Para
```

Donde resto(x, n) es el resto de la división entera de “x” por “n”. El pseudicódigo para el cálculo de esta operación se muestra a continuación:

Cálculo del resto de la división entera de “x” por “n”

```
Resto : = x
Si x ≥ 0 entonces
    Mientras resto ≥ n hacer
        resto : = resto – n
    Fin Mientras
Sino
    Mientras resto < 0 hacer
        resto : = resto + n
    Fin Mientras
Fin Si
```

El siguiente ejemplo ilustrará cómo se calcula la potencia modular mediante este algoritmo.

Ejemplo 5.1

Calcular $13^{40} \pmod 9$ trabajando con 8 bits.

Solución

En este caso, x = 13 , a = 40 , n = 9 y L = 8. Primeramente se expresa a= 40 en base 2, utilizando L= 8 dígitos binarios:

$$40 = 0\ 0\ 1\ 0\ 1\ 0\ 0\ 0$$

La siguiente tabla muestra los cálculos efectuados al implementar el algoritmo, aplicando las propiedades enunciadas en la Proposición 1.4:

i	a_i	$x^{2^i} = 13^{2^i} \pmod 9$	**Potencia modular**
0	0	$13^{2^0} = 13 \equiv 4$	1
1	0	$4^2 = 16 \equiv 7$	1
2	0	$7^2 = 49 \equiv 4$	1
3	1	$4^2 = 16 \equiv 7$	7 (1) = 7
4	0	$7^2 = 49 \equiv 4$	7
5	1	$4^2 = 16 \equiv 7$	7 (7) = 49 ≡ 4
6	0	$7^2 = 49 \equiv 4$	4
7	0	$4^2 = 16 \equiv 7$	4

Por lo tanto, $13^{40} \equiv 4 \pmod 9$ ■

5.2 Espacio de memoria y tiempo requerido para el cálculo de la exponenciación modular

Primeramente se determinará el tiempo requerido para computar $x^a \pmod n$. En general, un algoritmo ejecuta aproximadamente la misma cantidad de sumas y restas que de multiplicaciones y divisiones. Por esta razón sólo se contarán las multiplicaciones y divisiones que lleva a cabo el algoritmo de exponenciación modular.
Al analizar cada línea del pseudocódigo resulta:

```
Potencia = 1
x : = resto(x, n)          No hay multiplicaciones ni divisiones pues son
                           sucesivas restas.
Si a0 = 1 entonces
Potencia : = x
Fin Si
Para i = 1  hasta L - 1 hacer
   x : = x^2               L^2 multiplicaciones pues x tiene L-dígitos binarios.
   x : = resto(x, n)       No hay multiplicaciones ni divisiones.
   Si ai = 1 entonces
   Potencia : = Potencia · x   L^2 multiplicaciones.
   Fin Si
Fin Para
```

Por lo tanto, para cada i= 1, .., L- 1, se efectúan $2\,L^2$ multiplicaciones. En consecuencia, el total de operaciones aritméticas es $2\,L^2\,(L-1) = 2\,L^3 - 2\,L^2 = O(L^3)$.
De este modo queda demostrado que el algoritmo requiere $O(L^3)$ tiempo de ejecución.

A continuación se muestra el espacio de memoria que requiere el algoritmo. Para ello se analizará la cantidad de bits que ocupa cada variable que interviene en el pseudocódigo, teniendo en cuenta que “x”, “n” y “a” ocupan L-bits:

2 L bits para la variable Potencia
L bits para la variable x
L bits para la variable resto(x, n)
L bits para los dígitos binarios a_i, i = 0, …, L - 1
2 L bits para la variable x^2

Luego, se utiliza 2 L bits + L bits + L bits + L bits + 2 L bits = 7 L bits = O(L) bits.
Por lo tanto, queda probado que el algoritmo necesita O(L) bits de memoria.

5.3 *Diseño de la compuerta cuántica para el cálculo de la exponenciación modular*

El pseudocódigo que permite calcular $x^a \pmod n$ puede ser implementado fácilmente mediante una compuerta cuántica. La única "parte complicada" es cuando se calcula el producto, como ocurre en las líneas 7 ($x := x^2$) y 10 (Potencia := Potencia $\cdot$ x).
Por esta razón sólo se mostrará cómo diseñar una compuerta reversible que permita calcular el producto modular de dos números naturales "b" y "c". Dicha compuerta tendrá por entrada al número "b" y como salida al producto "b c (mod n)". En este caso, los números "c" y "n" se definen dentro de la estructura de la compuerta.
Para que dicha compuerta sea reversible, deberá ser capaz de recuperar "b", lo cual será posible si existe $c^{-1} \pmod n$. Por otro lado se sabe, por Teorema 1.1, que $c^{-1} \pmod n$ existe si y sólo si mcd(c, n) = 1. La siguiente figura muestra un bosquejo del accionar de la compuerta:

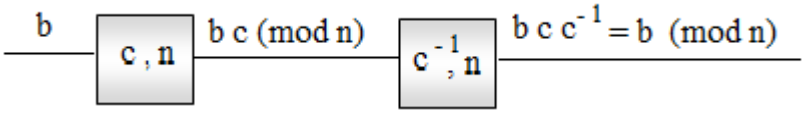


Figura 5.1: Bosquejo de una compuerta cuántica que calcula bc (mod n)

La compuerta se diseña utilizando dos subrutinas:

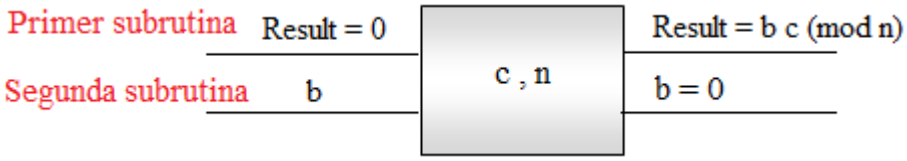


Figura 5.2: Compuerta cuántica que calcula bc (mod n)

Primer Subrutina:

Consiste en calcular el producto b c. Para ello, si consideramos la representación binaria de "b", es decir,

$$b = b_{L-1}\, b_{L-2} \ldots b_1\, b_0$$

Entonces

$$b = \sum_{i=0}^{L-1} b_i\, 2^i$$

Por lo tanto

$$b\, c = \sum_{i=0}^{L-1} b_i\, 2^i\, c$$

En este resultado se basa el pseudocódigo para el cálculo de bc, que es el siguiente:

Subrutina 1

```
Result := 0
Para i = 0 hasta L - 1 hacer
    Si b_i = 1 entonces
    Result : = Result +  2^i c
    Fin Si
Fin Para
```

Aquí, 2^i c puede ser calculado dentro de la estructura del pseudocódigo, teniendo en cuenta que si "c" está expresado en binario, 2^i c agrega "i" ceros a la derecha de c.

Segunda subrutina

Consiste en hacer b = 0. Este paso es necesario a fin de que la compuerta sea reversible. Para ello se debe considerar que, como b c quedó almacenado en la variable Result, y si $Result_i$ son los dígitos binarios de Result entonces

$$b\ c = Result = Result_{L-1}\ Result_{L-2}\ ...\ Result_1\ Result_0$$

Entonces

$$b\ c = \sum_{i=0}^{L-1} Result_i\ 2^i$$

De aquí se obtiene

$$b\ c\ c^{-1} = \sum_{i=0}^{L-1} Result_i\ 2^i\ c^{-1}$$

Por lo tanto

$$b = \sum_{i=0}^{L-1} Result_i\ 2^i\ c^{-1}$$

En consecuencia,

$$b - \sum_{i=0}^{L-1} Result_i\ 2^i\ c^{-1} = 0$$

Se ha recurrido a este artificio matemático debido a que no se puede asignar directamente a "b" el valor "0", pues en ese caso no se tendría una operación reversible, y la reversibilidad es absolutamente necesaria en una compuerta cuántica.
En base a este resultado, el pseudocódigo para transformar b en 0 es el siguiente:

Subrutina 2

```
Para i = 0 hasta L - 1 hacer
    Si Result_i = 1 entonces
    b: = b -  2^i  c^-1
    Fin Si
Fin Para
```

Verificación de la reversibilidad de la compuerta

Ahora, si se aplican los valores de salida (b c , 0) a la misma compuerta pero con c^{-1} en lugar de c, se podrán recuperar los valores originales de las variables Result y b:

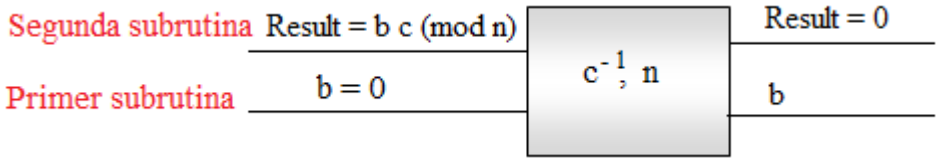


Figura 5.3: Reversibilidad de la compuerta cuántica que calcula bc (mod n)

Para recuperar "b"

Como

$$b\,c = \sum_{i=0}^{L-1} Result_i \; 2^i$$

Entonces

$$b = \sum_{i=0}^{L-1} Result_i \; 2^i \; c^{-1}$$

Luego, para recuperar "b" se aplica la Subrutina 1 con "Result" en lugar de "b" y "b" en lugar de "Result". Es decir,

b:= 0
Para i = 0 hasta L - 1 hacer
  Si $Result_i$ = 1 entonces
  b: = b + 2^i c^{-1}
  Fin Si
Fin Para

Para recuperar en Result el valor "0"

Como

$$Result = b\,c = \sum_{i=0}^{L-1} b_i \; 2^i \; c$$

Entonces

$$Result - \sum_{i=0}^{L-1} b_i \; 2^i \; c = 0$$

Es decir,

$$Result - \sum_{i=0}^{L-1} b_i \; 2^i \; \left(c^{-1}\right)^{-1} = 0$$

Luego, para asignar a Result el valor 0, se aplica la Subrutina 2 con “Result” en lugar de “b” y “b” en lugar de “Result”. Es decir,

Para i = 0 hasta L - 1 hacer

Si $b_i = 1$ entonces

Result: = Result - $2^i \left(c^{-1}\right)^{-1}$

Fin Si

Fin Para

Capítulo 6

Conclusiones

Capítulo 6: Conclusiones

La Computación Cuántica se encuentra en los inicios de su desarrollo y su marco teórico se basa en los Principios de la Mecánica Cuántica. El presente trabajo de tesis brinda un enfoque matemático, con demostraciones y justificaciones detalladas, no sólo de las propiedades cuánticas que se aplican para analizar el Algoritmo de Shor, sino también del comportamiento de las partículas subatómicas. El rigor matemático empleado facilitará al entendimiento de estos fenómenos tan complejos y servirá de complemento para otros temas de Física Cuántica.

Referencias

[1] Peter W. Shor. *Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer* . SIAM J.Sci.Statist.Comput, 1997.

[2] Luis Santaló . *Vectores y Tensores*. Ed. Eudeba,1961.

[3] Y. A. Kitaev. *Quantum Computing*. Springer Verlag, 2002.

[4] Juan Pedro Hecht. Fundamentos de Computación Cuántica**.** UBA, 2006.

[5] David Pérez. *Introducción a la Computación Cuántica para Ingenieros*. Ed. Alfaomega, 2013.

[6] http://www.fisicafundamental.net/misterios/computacion.html#introduccion

[7] D. Deutch and R. Jozsa. *Rapid solution of problems by quantum computation. Proc. Roy Soc. London Ser.* pp 553-558, 1992

[8] Vicente Moret Bonillo. *Principios Fundamentales de Computación Cuántica*. Ed. Universidad de La Coruña, 2013