UNIVERSIDAD DE GRANADA

TRABAJO FIN DE GRADO

INGENIERÍA EN INFORMÁTICA

Una singular forma de computar: los sistemas cuánticos

Autor
Francisco Amor Roldán

Director
Alberto Prieto Espinosa

Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación

Granada 3, septiembre de 2020

Una singular forma de computar: los sistemas cuánticos

Francisco Amor Roldán

Palabras clave: Computación Cuántica, Modelos de computación, Algoritmo de Grover, Física cuántica, Teletransporte, Información, IBM Quantum Experience, Qiskit, Ordenador cuántico.

Resumen

Este trabajo es un estudio introductorio en el que se exploran las posibilidades reales y prácticas, en el momento actual, de la computación cuántica en aplicaciones que son de incidencia en las telecomunicaciones y las ingenierías.

En primer lugar se realiza una exposición clara y rigurosa de los principios de la física y computación cuántica. Seguidamente se presentan protocolos y algoritmos de gran relevancia como es el caso del teletransporte, que es el principal método de transmisión de información cuántica, la codificación superdensa, con la que transmitir información clásica usando canales seguros de información cuánticos, o el algoritmo de búsqueda de Grover, que supone una mejora de los algoritmos clásicos de optimización.

En segundo lugar, se presenta la implementación como circuito de estos protocolos y algoritmos en tres ordenadores cuánticos reales proporcionados por la herramienta *IBM Quantum Experience*. Tras exponer cada implementación, se realiza una comparación de las prestaciones de estos dispositivos durante la ejecución del circuito atendiendo a las características técnicas de cada circuito y dispositivo. Para finalizar se presenta una evaluación global de los resultados obtenidos.

Una singular forma de computar: los sistemas cuánticos

Francisco Amor Roldán

Keywords: Quantum computation, Computer models, Grover's algorithm, Quantum physics, Teleportation, Information, IBM Quantum Experience, Qiskit, Quantum computer.

Abstract

This document is a introductory study that explores the real and practical posibilities, at the present time, of quantum computing in applications that are relevant to telecommunications and engineering.

Firstly, a clear and rigorous exposition of the principles of quantum physics and quantum computing is made. After it, protocols and algorithms of great relevance are presented, such as teleportation, which is the main method of transmission for quantum information, superdense coding, which is used to transmit classical information using secure quantum channels, or Grover's search algorithm, which is an improvement on the classic optimization algorithms.

Secondly, the implementation as a circuit of these protocols and algorithms in three real quantum computers provided by *IBM Quantum Experience* tool is presented. After exposing each implementation, a comparison of the performance of these devices during the execution of the circuit is made, taking into account the technical characteristics of each circuit and device. Finally, a global evaluation of the results obtained is presented.

Yo, **Francisco Amor Roldán**, alumno de la titulación TITULACIÓN de la **Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación de la Universidad de Granada**, con DNI 46070509Z, autorizo la ubicación de la siguiente copia de mi Trabajo Fin de Grado en la biblioteca del centro para que pueda ser consultada por las personas que lo deseen.

Fdo: Francisco Amor Roldám

Granada a 3 de septiembre de 2020 .

D. **Alberto Prieto Espinosa**, Profesor del Área de arquitectura de computadores del Departamento de arquitectura y tecnología de computadores de la Universidad de Granada.

Informa:

Que el presente trabajo, titulado ***Una singular forma decomputar: los sistemas cuánticos***, ha sido realizado bajo su supervisión por **Francisco Amor Roldán**, y autorizamos la defensa de dicho trabajo ante el tribunal que corresponda.

Y para que conste, expiden y firman el presente informe en Granada a 3 de septiembre de 2020 .

Los directores:

Alberto Prieto Espinosa

Agradecimientos

Transmitir mi más sincero agradecimiento a todos aquellos que me han ayudado a lo largo de la realización de este trabajo.

En primer lugar, a mi tutor, Alberto Prieto Espinosa, por su gran ayuda en la planificación, información y presentación de este Trabajo de Fin de Grado.

En segundo lugar, a mi familia, por su apoyo y consejos. A mi gran amiga Ana Ruiz de Santa Quiteria y su gato Ivy, por acogerme en su casa mientras redactaba el trabajo durante la cuarentena de la primavera de 2020. A mi gran amigo Edu González, por instruirme en física cuántica mientras cruzábamos el Atlas. A mi gran amigo Sergio Heredia por ayudarme tanto durante la carrera. Por último a mi amigo del grupo de Telegram de la ETSIT, Taxo Rubio, por ayudarme a utilizar Latex.

Índice general

Parte I

INTRODUCCIÓN

Capítulo 1

Motivación y objetivos del trabajo

En este primer capítulo del apartado I. INTRODUCCIÓN de la presente memoria, en primer lugar (Secc. 1.1) se describen los motivos que han llevado a la realización del trabajo que se presenta, haciendo referencia a la visión conceptual de la materia como contenedora de información y a los sistemas físicos como procesadores de ella. Los ordenadores convencionales se basan en las dinámicas de la física clásica que siguen los circuitos electrónicos, para procesar información codificada en un alfabeto binario (bits) y representada con magnitudes físicas tales como niveles de tensión, de corriente, estado de magnetización o nivel de intensidad luminosa. No obstante, existen otros modelos computacionales como los basados en membranas celulares, en cadenas de ADN o, como se expone en este trabajo, en sistemas cuánticos. Los sistemas cuánticos como procesadores de la información están siendo objeto de gran interés en las comunidades científicas y tecnológicas por su potencialidad en la resolución en tiempos increíblemente bajos de problemas específicos del ámbito de las ingenierías irresolubles incluso con los supercomputadores más potentes de la actualidad.

La Secc. 1.2 establece los objetivos generales y específicos del trabajo desarrollado. En definitiva el objetivo fundamental es explorar las posibilidades reales y prácticas, en el momento actual, de la computación cuántica en aplicaciones que son de incidencia en las telecomunicaciones y las ingenierías en general.

El capítulo concluye con la Secc. 1.3 que describe la organización de la presente Memoria.

1.1. Introducción y motivación

La información es lo que queda cuando uno se abstrae de los aspectos materiales de la realidad física. Aún siendo un concepto que habita en las descripciones más abstractas de la realidad, la información es consecuencia de ella.

El estudio de la información está ligado a avances en la física, y se remonta a conceptos ya descritos en las leyes de la termodinámica. La termodinámica es el estudio del comportamiento colectivo de entidades en una escala macroscópica, haciendo una descripción estadística del estado de estas entidades a nivel microscópico. El estado microscópico se corresponde con el movimiento de estas entidades, y el estado macroscópico se representa con la presión, temperatura y volumen de sistemas como gases y fluidos. La segunda ley de la termodinámica hace referencia a la entropía, y postula, que en un sistema termodinámico cerrado la entropía tiende a incrementar con el tiempo hasta alcanzar un valor máximo.

La entropía es una magnitud que mide el grado de desorden en la configuración de los estados que forman un sistema. Cuanta más incertidumbre haya sobre el estado de estas entidades, mayor será la entropía. Este concepto está directamente relacionado con el de información: cuanto mayor sea la entropía de un sistema, más información será necesaria para describir los estados de sus entidades, es decir, si la entropía es alta, el sistema nos proporciona poca información. Para sistemas poco entrópicos, como los cristales, con unos pocos bits se puede describir la estructura que subyace a la configuración de los estados del sistema, el sistema proporciona mucha información. Autores como Seth Lloyd [11], hacen una interpretación de este concepto, como información invisible de un sistema, es decir, los sistemas físicos, tienen una cierta cantidad de información visible e invisible (entropía), y la dinámicas de estos sistemas procesan y transforman esta información, pero nunca la destruyen. En este sentido, podemos ver la materia como contenedora de información y a los sistemas físicos como procesadores de ella.

Desde este punto de vista, podemos entender que existe una relación entre un sistema físico capaz de transformar materia, y un ordenador. Clare Horseman en el artículo "When does a physical system compute?"[8], propone un esquema, en el que la propia interpretación humana de estos sistemas, delimita el concepto de computador. En este esquema (ver Figura 1.1), un problema abstracto P, se formaliza dentro del modelo matemático de un sistema físico m, este modelo ha de ser codificado en el sistema físico real ,S, que evoluciona según sus propias dinámicas ,$D(S)$, hasta llegar al estado S'. Llegado este punto, es necesario realizar una decodificación de la información contenida en este sistema, y obtener el resultado de nuestra computación en el marco abstracto de nuestro modelo del sistema.

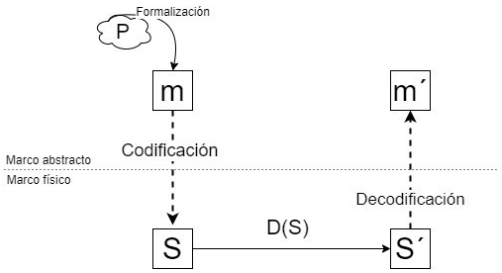


Figura 1.1: Diagrama de la interpretación de un sistema físico como ordenador.

Los ordenadores convencionales, se basan en las dinámicas de la física clásica que siguen los circuitos electrónicos para procesar información en forma de bits, estos bits toman el valor de 0 ó 1 en función de dos niveles posibles en magnitudes físicas tales como tensión, corriente, magnetización o intensidad luminosa. Existen muchos otros modelos de computación no convencionales, como computación basada en membranas celulares, en cadenas de ADN, o como se expone en este trabajo, en sistemas cuánticos.

La computación convencional se desarrolla en un contexto que podemos denominar "macroscópico". Experimentamos, entendemos e intuimos lo que ocurre en el mundo macroscópico, sin embargo, no conocemos, ni experimentamos directamente los mundos microscópico y "nanoscopico". Esto hace que sea difícil de entender la Física Cuántica, y la encontremos misteriosa o exótica. Aparecen propiedades y fenómenos muy alejados de nuestras experiencias directas y de nuestra intuición. Por ejemplo, en el mundo cuántico una partícula puede atravesar sin mayor problema una pared u obstáculo, o se presenta la paradoja del gato de Schröedinger, que encerrado en una caja puede llegar a estar a la vez vivo y muerto. Aunque las propiedades de la Física Cuántica nos parezcan muy extrañas, tienen un fundamento científico riguroso basado en teorías que han justificado satisfactoriamente fenómenos no explicables con la mecánica clásica tales como: el efecto fotoeléctrico, la radiación del cuerpo negro, el espectro atómico del átomo de hidrógeno, etc.

Teóricamente se ha probado que existen problemas del mundo real de gran complejidad, o incluso inabordables para los computadores convencionales que pueden ser resueltos satisfactoriamente con la computación cuántica. Debido a ello, el interés por este nuevo concepto ha ido incrementándose notablemente, llegando hasta más de 344.000 referencias a publicaciones y trabajos sobre computación cuántica entre 2014-2020 en Google académico. También es de mucho interés para grandes empresas tecnológicas como Google, Intel o IBM, que en los últimos años se han ido sumado a la carrera por la computación cuántica.

Este interés creciente por la computación cuántica, es el motivo que nos ha llevado a realizar una incursión en este campo y explorar sus posibilida-

des reales y prácticas, en el momento actual, en aplicaciones de incidencia en las telecomunicaciones como puede ser el teletransporte de información cuántica, el desarrollo de nuevos protocolos de codificación usando canales seguros de información cuántica, o la mejora de los algoritmos clásicos de optimización.

1.2. Objetivos del trabajo

El presente trabajo se enmarca en el contexto de la resolución de problemas específicos en el ámbito de las ingenierías. Sus objetivos, generales y específicos, son los siguientes:

1. Realizar una introducción a los principios de la computación cuántica necesarios para entender el resto de la presente memoria sin tener necesidad de un conocimiento previo sobre el tema o de acudir a fuentes que se encuentran dispersas en la bibliografía.

    1.1. Realizar una introducción a los principios matemáticos de la física y la computación cuántica.

2. Estudiar la viabilidad de la computación cuántica en la resolución problemas en el ámbito de las telecomunicaciones y las ingenierías en general.

    2.1. Presentar el protocolo de teletransporte como método de transmisión de información cuántica.

    2.2. Presentar el protocolo de codificación superdensa como método de transmisión de información clásica usando canales seguros de información cuánticos.

    2.3. Presentar el algoritmo de búsqueda de Grover como mejora de los algoritmos de búsqueda clásicos.

3. Programar los ordenadores cuánticos que ofrece la herramienta de IBM Quantum Experience y estudiar su viabilidad en el momento actual para la resolución de problemas reales en el ámbito de las telecomunicaciones y las ingenierías en general..

    3.1. Presentar las características generales de los ordenadores cuánticos, y el lenguaje de programación, Qiskit, que ofrece la herramienta IBM Quantum Experience.

    3.2. Estudiar la implementación de los algoritmos anteriores en ordenadores cuánticos reales.

    3.3. Evaluar los resultados obtenidos tras la ejecución de los algoritmos cuánticos en diferentes ordenadores, comparar sus prestaciones y delimitar el rango de uso de estos dispositivos a día de hoy.

1.3. Organización de la presente memoria

Esta Memoria, que presenta el desarrollo y resultados obtenidos en el trabajo desarrollado como proyecto Fin de Grado, se ha organizado en cuatro grandes apartados: I. Introducción, II. Marco teórico, III. Resolución del Trabajo, y IV. Conclusiones y vías futuras. Por último se presenta la bibliografía utilizada.

El Apartado I., es el presente y está dedicado a describir los motivos que han llevado a realizar este trabajo y a los objetivos establecidos.

El apartado II. está constituido por dos capítulos. El primero (Capítulo 2) se dedica a presentar unas nociones de física cuántica. Se trata de realizar una introducción a los principios de la computación cuántica imprescindibles para entender el resto de la presente Memoria sin tener necesidad de un conocimiento previo sobre el tema o de acudir a fuentes que se encuentran dispersas en la bibliografía. El segundo (Capítulo 3) trata de recopilar de forma clara y rigurosa, a la vez que concisa, la aplicación de los sistemas cuánticos a la computación, mostrando el origen de la gran potencia de cálculo obtenible con ellos.

El apartado III. está formado por dos capítulos. El primero de ellos (Capítulo 4) describe los recursos utilizados para la realización de los experimentos desarrollados: características de los computadores cuánticos utilizados y herramientas usadas para la redacción de los programas cuánticos que se presentan. El segundo (capítulo 5), muestra los resultados obtenidos en la ejecución de cuatro aplicaciones que se han considerado de interés en el ámbito de la ingeniería, y de las telecomunicaciones en particular.

El Apartado IV. está constituido por un único capítulo (Capítulo 6) dedicado a analizar la consecución de los objetivos planteados y las posibles vías de trabajo en un futuro.

Por último, se presenta la bibliografía utilizada para el desarrollo del trabajo presentado y para la redacción de la presente Memoria.

Parte II

MARCO TEÓRICO

Capítulo 2

Nociones de Física cuántica

"Quantum mechanics" is the description of the behavior of matter and light in all its details and, in particular, of the happenings on an atomic scale. Things on a very small scale behave like nothing that you have any direct experience about. They do not behave like waves, they do not behave like particles, they do not behave like clouds, or billiard balls, or weights on springs, or like anything that you have ever seen. ("The Feynman Lectures on Physics Vol. III Ch. 1: Quantum Behavior", 1965) [7]

En este capitulo se exponen los principios de la física cuántica necesarios para entender el presente trabajo. En la Secc. 2.1, se presenta el experimento de la doble rendija, que aporta una visión general y didáctica del comportamientos de los sistemas cuánticos. En la Secc. 2.2, se explican los tres fundamentos básicos de la física cuántica: la superposición, la medida y la evolución de los sistemas cuánticos. Por último, en la secc. 2.3, se presenta el qubit, la unidad básica de información cuántica.

2.1. Experimento de la doble rendija

Realizado por primera vez por Thomas Young en 1803, el experimento de la doble rendija, fue un intento de discernir entre la naturaleza corpuscular(localizable) u ondulatoria de la luz. En el experimento se lanzan partículas desde una fuente hacia una placa con dos rendijas. Tras la primera placa, hay una segunda placa paralela a la primera, que dispone de un detector para capturar la posición de esta donde colisiona la partícula.

El experimento ilustra el comportamiento cuántico de las partículas de manera experimental, y da luz a conceptos como: la superposición de estados, el uso de probabilidades para describir los sistemas, la perturbación de los sistemas al ser medidos o el principio de incertidumbre de Heisenberg.

El experimento se describirá brevemente con electrones, y para entender el comportamiento cuántico de los electrones, se comparará con el comportamiento corpuscular (una bala) y el ondulatorio (olas).

2.1.1. Experimento con Balas

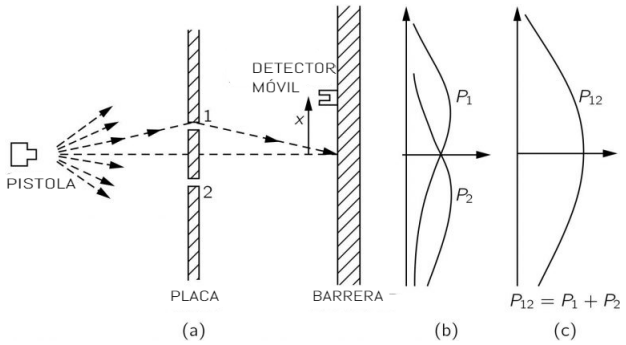


Figura 2.1: Experimento de la doble rendija con balas.

En este caso tenemos una pistola que dispara balas con trayectoria aleatoria. Estas balas son indestructibles, y nuestro detector las detecta enteras. Se describe el sistema en la figura 2.1.

Si cerramos la rendija 2, la distribución de probabilidad de colisión en un punto x de la segunda placa es P_1. Lo mismo pasa con P_2 si cerramos la primera rendija. En el caso de que ambas rendijas estén abiertas, la distribución es $P_{12} = P_1 + P_2$. Por tanto, no hay interferencias entre P_1 y P_2.

2.1.2. Experimento con olas

En el movimiento ondulatorio, en vez de hablar de probabilidad, hablaremos de intensidad. El detector captura la intensidad de la onda en un punto x de la segunda placa. El experimento se describe de la misma forma salvo que se restringe la reflexión de la onda en la segunda placa ("Absorber").

En este caso se puede comprobar que la intensidad cuando ambas rendijas están abiertas I_{12} no es igual a $I_1 + I_2$, hay interferencias entre ellas. Esto se debe a que la intensidad es proporcional a la amplitud de onda h, $I = |h|^2$, por tanto $I_{12} = |h_{12}|^2$ y $h_{12} = h_1 + h_2$. Nos queda $I_{12} = |h_1 + h_2|^2$.

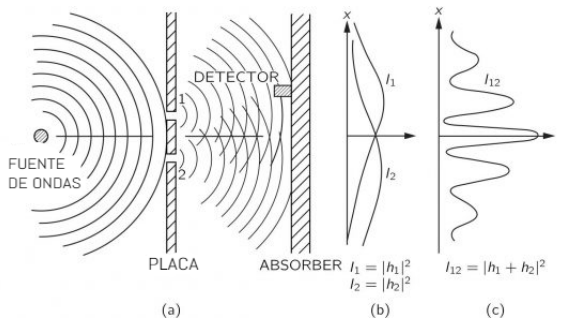


Figura 2.2: Experimento de la doble rendija con olas.

2.1.3. Experimento con electrones

En este caso tenemos una fuente de electrones que emite de forma discreta y al igual que las balas, los electrones llegan "enteros" al detector. Cabe destacar que en todas las variantes del experimento solo obtenemos información gracias al detector, no observamos lo que está pasando.

Podemos ahora hacer la siguiente proposición:

"Cada electrón pasa por la rendija 1 o la rendija 2".

Entonces los electrones serán de dos clases: a) los que pasaron por la rendija 1, b) los que pasaron por la rendija 2. Se describen P_1 y P_2 igual que en el primer caso de las balas. Pero para el caso en el que ambas rendijas están abiertas P_{12}, observamos un patrón de interferencias, al igual que con las olas. Para los electrones $P_{12} \neq P_1 + P_2$.

Para describir este fenómeno, introducimos ϕ, la amplitud de probabilidad. La amplitud de probabilidad es una función de x, y el sistema se comporta de la misma forma que en el caso de las olas.

$P = |\phi|^2$, $P_{12} = |\phi_{12}|^2$, $\phi_{12} = \phi_1 + \phi_2$ y finalmente, $P_{12} = |\phi_1 + \phi_2|^2$.

Podemos concluir en que los electrones llegan en paquetes, como partículas, pero la probabilidad de llegada está distribuida como la intensidad de una onda. Por tanto, como el número de electrones que llegan a un punto particular no es igual al número de electrones que llegan pasando por la rendija 1 más los que pasan por la rendija 2 (porque tenemos un patrón de interferencias), podemos concluir que nuestra proposición anteriormente formulada es falsa. No es verdad que cada electrón pase por la rendija 1 o la rendija 2. El sistema se describe como una superposición de ambas posibilidades.

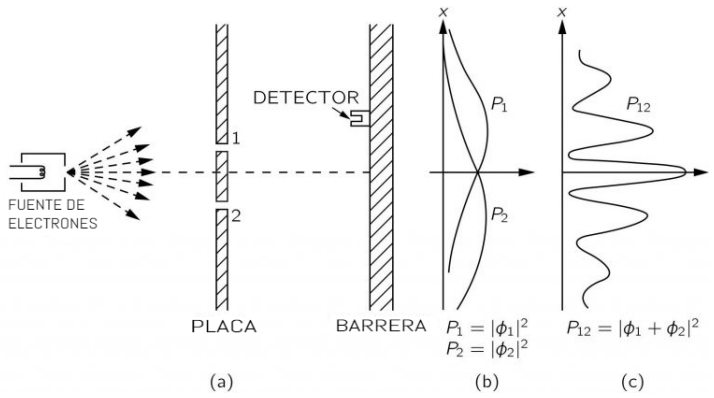


Figura 2.3: Experimento de la doble rendija con electrones.

2.1.4. Experimento con electrones, observando

En este caso, observamos por qué rendija pasa cada electrón mientras se realiza el experimento, y para ello medimos con una fuente de luz por qué rendija pasa el electrón. El sistema, bajo medida, se comporta exactamente igual que en el caso de las balas: $P'_{12} = P'_1 + P'_2$.

Al realizar la medida, perturbamos el sistema, y la superposición de estados que se usaba para medir el sistema (en la que el electrón actuaba de manera ondulatoria) colapsa a estados concretos como un sistema corpuscular.

Con estos datos se puede formular el Principio de incertidumbre de Heisenberg para el experimento de la siguiente forma:
"Es imposible diseñar un aparato de medida que detecte por qué rendija pasa el electrón sin perturbar el patrón de interferencia del sistema"

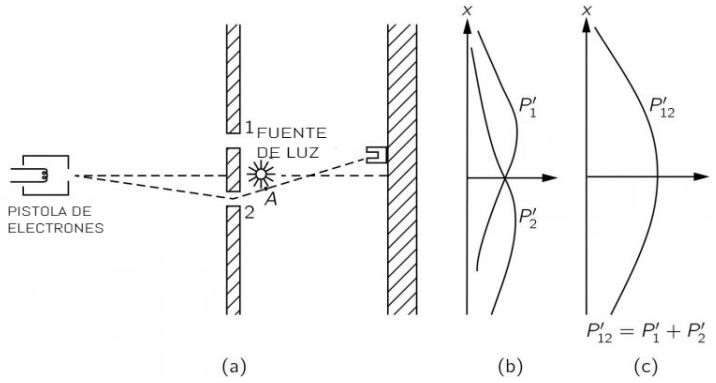


Figura 2.4: Experimento de la doble rendija con electrones, observando.

2.2. Fundamentos de la física cuántica

En el apartado anterior se mostró brevemente el comportamiento experimental de los sistemas cuánticos. En el presente apartado se exponen los conceptos fundamentales para poder describir estos sistemas, en definitiva se presenta el modelado matemático de la física cuántica [3].
Los sistemas cuánticos se pueden describir usando un espacio vectorial sobre los números complejos de n dimensiones. Para ello usaremos los *Espacios de Hilbert*. Un espacio de Hilbert es una generalización de un espacio vectorial euclideo de $n \in N$ dimensiones sobre C. Sin entrar en más detalle sobre estos espacios, cabe mencionar que se preservan conceptos como el producto escalar y la norma euclidea.

Para describir el espacio se suele usar la notación de Dirac o "Bra-ket". En esta notación los 'bra' $\langle B|$ representan vectores fila, y los 'ket' $|A\rangle$ vectores columna. El producto escalar se representa como $\langle B|A\rangle$.

$$|A\rangle = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} \quad \langle B| = \begin{pmatrix} b_1 & b_2 & b_3 \end{pmatrix}$$

2.2.1. Superposición de estados

Un sistema de n estados se puede describir por un espacio de Hilbert normalizado de n dimensiones H_n. Este sistema puede estar en superposición de varios estados, cada estado corresponde un vector perteneciente a una base ortonormal del espacio. El sistema en superposición se puede representar por un vector unitario expresado como una combinación lineal de los vectores de una base ortonormal de $H_n : |v_1\rangle, |u_2\rangle ..., |u_n\rangle$

$$|v\rangle = \alpha_1 |u_1\rangle + \alpha_2 |u_2\rangle ... + \alpha_n |u_n\rangle = \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ ... \\ \alpha_n \end{pmatrix} \in C^n$$

$$\sum_{i=1}^{n} |\alpha_i|^2 = 1.$$

Ahora $|v\rangle$ representa el estado del sistema como una superposición de los estados $|v_1\rangle, |u_2\rangle ..., |u_n\rangle$, y α_i representa la amplitud correspondiente al estado al estado $|u_i\rangle$.

Para un sistema de 2 estados descrito por H_2 , para la base

$$|e_1\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, |e_2\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

el sistema se describe por $|x\rangle$:

$$|x\rangle = w_1 |e_1\rangle + w_2 |e_2\rangle ,$$

$$|w_1|^2 + |w_2|^2 = 1.$$

Suponiendo que w_1, w_2 sean reales, se puede hacer un visualización didáctica de este sistema en el plano. Cada w_i puede expresarse en función del ángulo ϕ entre $|x\rangle$ y $|e_1\rangle$,

$$|x\rangle = cos\phi\, |e_1\rangle + sin\phi\, |e_2\rangle .$$

Y más generalmente, como nuestro espacio está normalizado (todos los vectores son unitarios, y por tanto el producto escalar entre dos vectores equivale al coseno del ángulo menor entre ellos) cada w_i puede definirse como la proyección de $|x\rangle$ sobre $|e_i\rangle$. El sistema se puede reescribir de la siguiente manera:

$$|x\rangle = \langle e_1|x\rangle\, |e_1\rangle + \langle e_2|x\rangle\, |e_2\rangle .$$

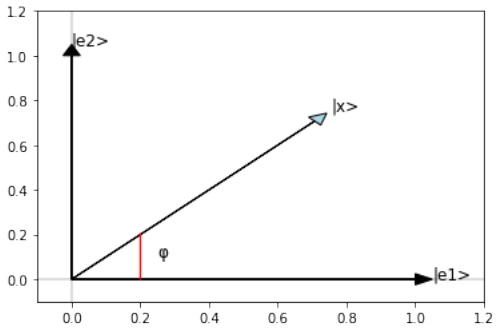


Figura 2.5: Sistema bidimensional en estado x.

2.2.2. Medida del sistema

Una medida se especifica eligiendo una base ortonormal del sistema y se dice que se mide sobre esa base. Los estados observables son los estados de la base.

La medida del sistema implica el colapso del vector de estado en superposición, que matemáticamente corresponde con una proyección sobre un vector de la base en la cual observamos en sistema. La probabilidad de obtener cada estado al medir viene dada por el cuadrado de la longitud de esa proyección. Se puede formular de la siguiente manera: Para un sistema en H_n usando la base

$$|v_1\rangle , |u_2\rangle ..., |u_n\rangle$$

y con un vector de estado

$$|v\rangle = \alpha_1 |u_1\rangle + \alpha_2 |u_2\rangle ... + \alpha_n |u_n\rangle$$

la probabilidad de obtener cada estado $i \in 1, 2, ...n$ al medir es

$$P[i] = |\langle u_i|v\rangle|^2 = |\alpha_i|^2.$$

Esta probabilidad es un numero real entre 0 y 1. Dado que

$$\sum_{i=1}^{n} |\alpha_i|^2 = 1.$$

Después de que un estado $|u_i\rangle$ sea observado(medido) el sistema se encuentra en el estado básico

$$|v\rangle = 1 \cdot |u_i\rangle .$$

Observables

La medida se determina por la base que escojamos, y como nuestro vector se puede expresar en diferentes bases, podemos hacer diferentes medidas del sistema. Un observable es una cantidad o propiedad que podemos medir en un sistema, este se puede describir dando una base del espacio o la matriz de densidad del observable. Entonces el observable puede ser descrito por un operador (matriz) hermítico de nuestro espacio.

Según el *teorema espectral*, una matriz $A_{k\times k}$ de estas características, A tiene un conjunto de autovectores $|a_1\rangle , |a_2\rangle , ... |a_k\rangle$ con autovalores $\lambda_1, \lambda_2, ...\lambda_k$ que forman una base ortonormal del espacio.

Por tanto la matriz A define el observable A. Al realizar una medida sobre este observable, el sistema colapsa a uno de sus autovectores y obtenemos su autovalor como resultado. Para construir la matriz de densidad en función de la base y sus autovalores se introduce el concepto de operador de proyección P_x, para

$$|x\rangle = \begin{pmatrix} a \\ b \end{pmatrix}$$

$$P_x = |x\rangle \langle x| = \begin{pmatrix} |a|^2 & a \cdot \overline{b} \\ b \cdot \overline{a} & |b|^2 \end{pmatrix}$$

P_x es un operador del espacio que se usa para calcular la proyección de un estado arbitrario $|v\rangle$ sobre $|x\rangle$

$$P_x |v\rangle = |x\rangle \langle x| |v\rangle = |x\rangle \langle x|v\rangle = \langle x|v\rangle |x\rangle$$

donde $\langle x|v\rangle$ representa la amplitud de $|v\rangle$ sobre $|x\rangle$.
Entonces conociendo los autovectores $|a_1\rangle , |a_2\rangle , ... |a_k\rangle$ y autovalores $\lambda_1, \lambda_2, ...\lambda_k$ podemos construir A de la siguiente manera:

$$A = \sum_{i=1}^{n} \lambda_i P_i$$

Entropía de Von Neumann

La entropía de un sistema cuántico se mide de acuerdo con la definición de Von Neumann. Para un estado

$$|x\rangle = \begin{pmatrix} x_1 \\ x_2 \\ ... \\ x_n \end{pmatrix}$$

puede ser expresado como P_x, y P_x puede ser descompuesta en autovalores y autovectores al igual que la matriz A.

$$P_x = \lambda_1 |x_1\rangle \langle x_1| + \lambda_1 |x_2\rangle \langle x_1| + ... + \lambda_n |x_n\rangle \langle x_n|$$

La entropía del estado se define como

$$E(P_x) = -\sum_{i=1}^{n} (\lambda_i log \lambda_i)$$

$$\sum_{i=1}^{n} \lambda_i = 1$$

los estados más entrópicos serán los que tengan una superposición equiprobable y los que estén en estados puros tendrán entropía 0 (para el estado puro $i, \lambda_i = 1$ y el resto son 0. Por tanto $0 = 1 \cdot log \cdot 1$).

2.2.3. Evolución unitaria

La evolución temporal de un sistema cuántico cerrado (que no está en contacto con otros sistemas) es descrita por la *ecuación de Schrödinger*,

$$i\hbar \frac{d|v\rangle}{dt} = H|v\rangle ,$$

donde $i = \sqrt{-1}$, $\hbar$ es la constante de plank reducida, y H es un observable del sistema llamado Hamiltoniano cuyos autovectores son los estados propios del sistemas y sus autovalores los niveles de energía de cada estado.
Sin entrar en detalle, la evolución temporal de uno de los estados propios $|v(0)\rangle = |v_j\rangle$ con autovalor λ_j a un estado $|v(t)\rangle$, resolviendo la ecuación de Schrödinger, se puede expresar de la siguiente manera para $|v(t)\rangle$,

$$|v(t)\rangle = e^{\frac{-i\lambda_j t}{\hbar}} |v(0)\rangle$$

Para el propósito de este trabajo es suficiente con una descripción más general, y podemos formularla de la forma que se indica a continuación:

La evolución de un sistema cuántico cerrado se describe por una transformación unitaria, U. Es decir, el estado $|v_{t1}\rangle$ en en tiempo t_1, se relaciona

con el estado $|v_{t2}\rangle$ en t_2, mediante un operador unitario U, que representa la siguiente transformación:

$$|v_{t1}\rangle = U\,|v_{t2}\rangle\,.$$

Las matrices unitarias, son matrices complejas que satisfacen la condición de que $U^*U = UU^* = I$, donde I representa la matriz identidad y U^* es la matriz transpuesta conjugada de U. Sabiendo esto, podemos deducir que las operaciones en nuestro sistema son reversibles. La transformación unitaria, es una transformación lineal y conserva tanto la longitud de los vectores como los ángulos entre ellos.

Podemos entonces, ver las transformaciones como rotaciones de nuestro espacio. Se ilustra en el siguiente ejemplo:

Ejemplo 2.1 *Dado un sistema en H_2 en el estado $|v\rangle = \frac{1}{\sqrt{2}}|u_1\rangle + \frac{1}{\sqrt{2}}|u_2\rangle$, se le aplica un operador de rotación R_θ, con $\theta = -\frac{\pi}{2}$ (entre $|v\rangle$ y $|u_1\rangle$), R_θ queda*

$$R_\theta = \begin{pmatrix} cos\theta & sin\theta \\ -sin\theta & cos\theta \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

Ahora podemos aplicar el operador

$$R_\theta\,|v\rangle = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{pmatrix}$$

$$R_\theta\,|v\rangle = |v'\rangle = \frac{1}{\sqrt{2}}\,|u_1\rangle - \frac{1}{\sqrt{2}}\,|u_2\rangle$$

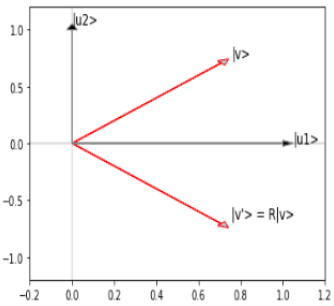


Figura 2.6: Visualización en el plano de $R_\theta\,|v\rangle = |v'\rangle$

2.3. El qubit

En computación cuántica la unidad básica de información es el qubit, un sistema cuántico de dos estados. El sistema se suele expresar en la base $|0\rangle, |1\rangle$ de la siguiente manera:

$$|v\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\,,$$

con

$$|\alpha|^2 + |\beta|^2 = 1.$$

Un qubit, al igual que un bit, es un concepto abstracto y no esta asociado a un sistema físico concreto. Hay varias formas de implementar un qubit con sistemas físicos como fotones, puntos cuánticos [4], electrones o núcleos atómicos.

2.3.1. El qubit como Spin

Una idea de implementación sencilla es asociar el valor de un qubit al estado del *spin* de un electrón. El spin es una propiedad física de las partículas elementales asociada con su momento angular intrínseco. Para un electrón existen dos posibles estados del spin denotados como 'arriba' y 'abajo': $|\uparrow\rangle, |\downarrow\rangle$. El spin puede estar en una superposición de estos estados y podemos expresar un qubit en función del spin de un electrón:

$$|v\rangle = \alpha\,|\uparrow\rangle + \beta\,|\downarrow\rangle\,.$$

El spin puede ser manipulado y medido con diferentes dispositivos. Un ejemplo sencillo puede ser el experimento de *Stern Gerlach*, en el que aplicando campos magnéticos sobre el electrón se desvía su trayectoria en función del estado inicial de su spin.

Los posibles estados del spin pueden ser visualizados mediante *la esfera de Bloch*. Esta es una representación tridimensional de espacios complejos de dos dimensiones. Se ilustra en la siguiente figura [16]:

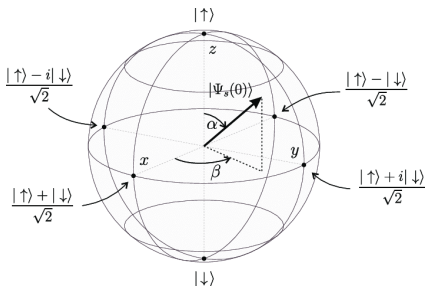


Figura 2.7: Estados del Spin.

2.3.2. Esfera de Bloch

La esfera de Bloch es una representación geométrica del espacio de estados de un sistema cuántico de dos niveles. Es una forma intuitiva de visualizar un qubit y las transformaciones que se aplican sobre él. Para construirla hay que representar las amplitudes del sistema en forma polar:

$$|v\rangle = \alpha\,|0\rangle + \beta\,|1\rangle = r_0 e^{i\phi_0}\,|0\rangle + r_1 e^{i\phi_1}\,|1\rangle\,,$$

donde α, β son números complejos y $re^{i\phi}$ su representación polar. Podemos extraer $e^{i\phi_0}$ como factor común

$$|v\rangle = e^{i\phi_0}[r_0\,|0\rangle + r_1 e^{i(\phi_1-\phi_0)}\,|1\rangle].$$

Como este coeficiente multiplica a todo el sistema, es físicamente irrelevante. Si se hace una medida del sistema, $e^{i\phi_0}$ no hará ninguna diferencia en el resultado. Entonces podemos expresar $(\phi_1 - \phi_0)$ como un único ángulo ϕ

$$|v\rangle = r_0\,|0\rangle + r_1 e^{i\phi}\,|1\rangle\,.$$

Sabiendo la condición de normalización

$$|r_0|^2 + |r_1|^2 = 1,$$

podemos expresar r_0, r_1 como $r_0 = cos\frac{\theta}{2},\ r_1 = sin\frac{\theta}{2}$ y nos queda

$$|v\rangle = cos\frac{\theta}{2}\,|0\rangle + sin\frac{\theta}{2}e^{i\phi}\,|1\rangle\,.$$

Por tanto nuestro espacio de dos dimensiones sobre números complejos puede ser descrito por dos parámetros reales θ y ϕ. Cada estado es un punto en la esfera.

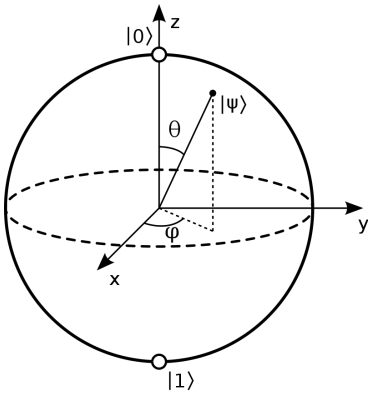


Figura 2.8: Estado ψ en esfera de Bloch.

En la esfera los estados $|0\rangle\,, |1\rangle$, son polos opuestos en el eje z.

Matrices de Pauli

Las matrices de Pauli Z, X, Y corresponden con observables en los diferentes ejes x, z, y de un qubit representado con la esfera de Bloch. Los resultados serán expresados en las bases $|0\rangle$ y $|1\rangle$, como es habitual en computación cuántica.

Para el eje z el estado apuntando a $+z$ es $|0\rangle$ con $\phi = 0$, y en su antípoda, apuntando a $-z$ se encuentra el estado $|1\rangle$ con $\phi = 2\pi$. Para esta base con autovalores $\lambda_0 = 1$ y $\lambda_1 = -1$, se construye el siguiente observable Z:

$$|0\rangle\,,\ \lambda_0 = 1$$
$$|1\rangle\,,\ \lambda_1 = -1$$
$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Para los estados en el eje x:

$$\theta = \frac{\pi}{2},\ \phi = 0.\ |v\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle\,,\ \lambda_0 = 1$$
$$\theta = \frac{\pi}{2},\ \phi = 2\pi.\ |v\rangle = \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle\,,\ \lambda_1 = -1$$
$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Para los estados en el eje y:

$$\theta = \frac{\pi}{2},\ \phi = \frac{\pi}{2}.\ |v\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{i}{\sqrt{2}}|1\rangle\,,\ \lambda_0 = 1$$
$$\theta = \frac{\pi}{2},\ \phi = \frac{-\pi}{2}.\ |v\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{-i}{\sqrt{2}}|1\rangle\,,\ \lambda_1 = -1$$
$$Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$$

Estas matrices son muy importantes porque dan las componentes de un estado, en las direcciones de los distintos ejes. Además en computación cuántica son de las matrices más utilizadas como operadores, constituyendo puertas lógicas en los 'circuitos' cuánticos.

Capítulo 3

Computación cuántica

En el capítulo anterior se describieron brevemente las dinámicas de la física cuántica. En este capítulo se expone como usar estas dinámicas para codificar, transformar y decodificar información. Una vez comprendemos cuál es el lenguaje de estos sistemas físicos, podemos empezar a hablar en él.

En la Secc. 3.1, se muestra el potencial que tienen los sistemas con qubits para codificar y transformar información. En la Secc. 3.2, se explica el entrelazamiento, un fenómeno cuántico que no tiene equivalente clásico. En la Secc. 3.3, se exponen el protocolo de teletransporte y de codificación superdensa, dos métodos de transmisión de información basados en el entrelazamiento cuántico. En la Secc. 3.4, se explica como implementar cualquier función clásica en un ordenador cuántico usando puertas lógicas universales y reversibles. Por último, en la Secc. 3.4, se presenta el algoritmo de Grover, un algoritmo de búsqueda que hace uso del gran potencial de computo paralelo que tienen los sistemas cuánticos en superposición.

3.1. Sistemas con qubits

El qubit (contracción de "quantum bit") es la unidad mínima de información en computación cuántica. Su análogo en computación clásica es el bit, y éste, como es bien conocido puede tomar sólo uno entre dos estados o valores: 0 ó 1. Sin embargo el qubit, que se rige por las leyes de la física cuántica, tiene dos estados fundamentales o básicos $|0\rangle$ o $|1\rangle$que corresponden al 0 y 1 clásicos, y que pueden estar en superposición coherente, es decir pueden ser 0 y 1 a la vez. Esto permite que con ellos puedan realizarse varias operaciones simultáneamente con esos dos valores [15].

3.1.1. Sistemas de un qubit

Un sistema de un qubit puede codificar $2^1 = 2$ estados. Su representación vectorial es la siguiente:

$$|v\rangle = \alpha\,|0\rangle + \beta\,|1\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}, \quad \text{para } |\alpha|^2 + |\beta|^2 = 1.$$

Normalmente, el qubit se expresa y se mide en las base $|0\rangle, |1\rangle$, se suele llamar base de 'bit' . Los estados más importantes son los estados

$$|0\rangle = 1 \cdot |0\rangle + 0 \cdot |1\rangle,$$

$$|1\rangle = 0 \cdot |0\rangle + 1 \cdot |1\rangle.$$

Aunque simbólicamente sean iguales, es importante ver la diferencia entre una base y un estado. La base, es el prisma con el que miras el sistema. Otros de los estados más importantes son los estados $|+\rangle, |-\rangle$:

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

$$|-\rangle = \frac{1}{\sqrt{2}}\,|0\rangle - \frac{1}{\sqrt{2}}\,|1\rangle.$$

Estos corresponden con los estados en el eje x de la esfera de Bloch y forman también una base ortonormal, se suele llamar 'base de signo'. Son estados muy interesantes porque forman una superposición equiprobable, es decir tienes 50 % de probabilidad de medir $|0\rangle$ y 50 % de medir $|1\rangle$. Los estados $|0\rangle, |1\rangle$ también se pueden expresar en base de signo haciendo una cambio de base:

$$|0\rangle = \frac{1}{\sqrt{2}}\,|+\rangle + \frac{1}{\sqrt{2}}\,|-\rangle,$$

$$|1\rangle = \frac{1}{\sqrt{2}}\,|+\rangle - \frac{1}{\sqrt{2}}\,|-\rangle.$$

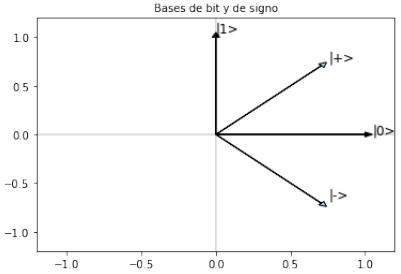


Figura 3.1: Representación en el plano de las bases de bit y signo.

Estas bases satisfacen la relación de incertidumbre al ser medidas, es decir, no puedes aumentar la certidumbre sobre una, sin disminuir la incertidumbre sobre otra. Para ilustrarlo podemos definir el concepto de 'extensión' de un estado sobre una base. La extensión sobre la base de signo se denota como $S_s(|v\rangle)$ y sobre el bit como $S_b(|v\rangle)$. Para un estado

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

$$S_b(|v\rangle) = |\alpha_0| + |\alpha_1|,$$

$$S_s(|v\rangle) = |\beta_0| + |\beta_1|.$$

Por ejemplo para $|0\rangle$ y $|+\rangle$:

$$S_b(|0\rangle) = 1,\ S_s(|0\rangle) = \sqrt{2}$$

$$S_b(|+\rangle) = \sqrt{2},\ S_s(|+\rangle) = 1$$

Entonces se puede formular el principio de incertidumbre de Heisenberg para las bases de bit y signo. Recordemos que al especificar una base, se definía un observable, una propiedad a medir del sistema. El principio se puede formular de la siguiente manera para un $|v\rangle$ arbitrario:

$$S_b(|v\rangle)S_s(|v\rangle) \geq \sqrt{2}.$$

Cuando aumentamos la certeza de obtener un valor al medir en una base, la disminuimos al medir en la otra.

Puertas lógicas cuánticas con un qubit

En computación cuántica, los operador unitarios actúan como puertas lógicas y definen la evolución del sistema. En este apartado se presentan algunas de las puertas más útiles. Se describe su efecto sobre un qubit arbitrario

$$|v\rangle = \alpha |0\rangle + \beta |1\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}.$$

Bit flip

Esta puerta intercambia las amplitudes de la base de un estado. Corresponde con la matriz de Pauli X.

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

$$X|v\rangle = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \beta \\ \alpha \end{pmatrix}$$

Para la base,

$$X|0\rangle = |1\rangle$$
$$X|1\rangle = |0\rangle$$

Phase flip

Esta puerta cambia el signo de la amplitud del segundo vector de la base. Corresponde con la matriz de Pauli Z.

$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$
$$Z|v\rangle = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \alpha \\ -\beta \end{pmatrix}$$

Para la base,

$$Z|0\rangle = |0\rangle,$$
$$Z|1\rangle = -|1\rangle.$$

Para los estados de signo,

$$Z|+\rangle = |-\rangle,$$
$$Z|-\rangle = |+\rangle.$$

Hadamard

Esta puerta, es clave en computación cuántica, puede pasar de un estado con mínima entropía a un estado con entropía máxima, es decir, una superposición equiprobable. Se cumple también que $H = H^*$.

$$H = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix}$$
$$H|v\rangle = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \frac{\alpha}{\sqrt{2}} \\ \frac{\beta}{\sqrt{2}} \end{pmatrix}$$

Para la base,

$$H|0\rangle = |+\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$$
$$H|1\rangle = |-\rangle = \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$$

Para los estados de signo,

$$H|+\rangle = |0\rangle, \quad H|-\rangle = |1\rangle.$$

Como podemos ver de un estado básico como $|0\rangle$, pasamos al estado $|+\rangle$, que contiene todos los posibles estados del sistema en una superposición equiprobable. Con una instancia de un qubit, podemos tener todos sus estados en paralelo. El problema es que para acceder a la información hay que realizar una medida, y la superposición colapsaría. Pero 'detrás del telón', en el mundo de de la física cuántica, al cual no podemos acceder, se presentan todas las posibilidades del sistema. Uno de los retos de la computación cuántica es investigar nuevas formas de acceder a la información que se presenta detrás del telón.

Las puertas X y Z pueden ser expresadas entre ellas:

$$X = HZH,$$

$$Z = HXH.$$

En el siguiente diagrama se puede ver como se relacionan los estados de bit y signo con estas puertas.

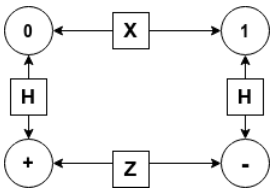


Figura 3.2: Relación entre estados y puertas lógicas.

3.1.2. Sistemas compuestos

Un sistema de dos qubits puede codificar $2^2 = 4$ estados. Para describir este sistema, hay que realizar una descripción conjunta de dos sistemas de un solo Qubit. Para ello utilizaremos el producto tensorial de ambos sistemas.

Para los estados de dos qubits

$$|v_1\rangle = \alpha_0 |0\rangle + \alpha_1 |1\rangle ,$$

$$|v_2\rangle = \beta_0 |0\rangle + \beta_1 |1\rangle .$$

El producto tensorial entre ellos es

$$|v_{12}\rangle = |v_1\rangle \otimes |v_2\rangle = \alpha_0\beta_0 |00\rangle + \alpha_0\beta_1 |01\rangle + \alpha_1\beta_0 |10\rangle + \alpha_1\beta_1 |11\rangle$$

Se puede hacer el producto tensorial de las bases de ambos sistemas y el resultado es la base de un espacio de Hilbert de 4 dimensiones H_4.

Generalmente el espacio generado por el producto tensorial de dos espacios H_n, H_w, es $H_n \otimes H_w = H_{n \cdot w}$. Para H_4 nos quedan la base (de bit)

$$|00\rangle = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \ |01\rangle = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \ |10\rangle = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \ |11\rangle = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}.$$

El producto tensorial para los operadores se describe la siguiente forma: Para A_{2x2}, B_{2x2}

$$A \otimes B = \begin{pmatrix} a_{11} \cdot B & a_{12} \cdot B \\ a_{11} \cdot B & a_{22} \cdot B \end{pmatrix} = W_{4x4}$$

Medida de sistemas compuestos

Sabiendo que un sistema de dos qubits se puede describir mediante el estado

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

La probabilidad de obtener un estado básico

$$|xy\rangle, \ x, y \in \{0, 1\}$$

viene dada por

$$P[|xy\rangle] = |\alpha_{xy}|^2.$$

Se puede hacen medidas parciales de un solo Qubit,

$$P[|x\rangle] = |\alpha_{x0}|^2 + |\alpha_{x1}|^2,$$

la probabilidad dependerá solo de la amplitud de ese Qubit. Despues de la medida, el estado resultante será normalizado y expresado en función de las bases que sean consistentes con el resultado de la medida. Digamos que para el sistema descrito por $|v\rangle$, hemos realizado una medida en el primer Qubit y obtenemos $|0\rangle$, el estado resultante será:

$$|v'\rangle = \frac{\alpha_{00} |00\rangle + \alpha_{01} |01\rangle}{\sqrt{|\alpha_{00}|^2 + |\alpha_{01}|^2}}$$

La medida para sistemas de dos qubits se puede generalizar para sistemas de n qubits.

Puertas lógicas cuánticas con sistemas compuestos

Como podemos usar el producto tensorial entre operadores, todos los operadores descritos para un Qubit, pueden aplicarse a sistemas de dos qubits. En general pueden aplicarse a sistemas de n qubits aplicando el producto tensorial n veces entre ellos. Por ejemplo una Hadamard para n qubits se define como $H \otimes H \otimes H ... \otimes H$ n veces, y se denota como $H^{\otimes n}$. Esto se aplica a cualquier operador o combinación de ellos.

Cnot

La puerta 'Cnot' o 'not controlada' es un operador de dos qubits, que implementa una puerta not sobre el segundo Qubit con el primer Qubit como controlador. Se puede visualizar con el siguiente diagrama de circuito.

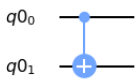


Figura 3.3: Representación de la puerta Cnot.

$$M_{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Para la base

$$M_{CNOT} |00\rangle = |00\rangle , M_{CNOT} |01\rangle = |10\rangle ,$$
$$M_{CNOT} |10\rangle = |11\rangle , \ M_{CNOT} |11\rangle = |10\rangle .$$

Para un estado $|v\rangle$ arbitrario,

$$|v\rangle = \alpha_{00} |00\rangle + \alpha_{01} |01\rangle + \alpha_{10} |10\rangle + \alpha_{11} |11\rangle$$
$$\downarrow M_{CNOT}$$
$$|v\rangle = \alpha_{00} |00\rangle + \alpha_{01} |01\rangle + \alpha_{11} |10\rangle + \alpha_{10} |11\rangle$$

Sistemas de n qubits

Un sistema de n qubits puede codificar 2^n estados básicos. El crecimiento del número de estados básicos en el sistema es exponencial:

- Un qubit $\in H_2, \ \alpha_0 |0\rangle + \alpha_1 |1\rangle .$
- Dos qubits $\in H_4, \ \alpha_{00} |00\rangle + \alpha_{01} |01\rangle + \alpha_{10} |10\rangle + \alpha_{11} |11\rangle .$

- Tres qubits $\in H_8$, $\alpha_{000}|000\rangle + \alpha_{001}|001\rangle + \alpha_{010}|010\rangle + \alpha_{011}|011\rangle + \alpha_{100}|100\rangle + \alpha_{101}|101\rangle + \alpha_{110}|110\rangle + \alpha_{111}|111\rangle$.
- n qubits $\in H_{2^n}$

$$\sum_{x=0}^{2^n} \alpha_x |x\rangle$$

El fenómeno es parecido al de los bits clásicos, el crecimiento del número de estados codificables es exponencial. Pero usando qubits, estos estados puede estar en superposición todos juntos en una sola instancia y podemos operar con ellos en paralelo. El reto sigue siendo que solo podemos acceder a un de ellos, con probabilidad

$$P[|x\rangle] = |\alpha_x|^2.$$

Al operar en un sistema de n qubits, las transformaciones afectan a todo el sistema. Si quisiéramos operar con un solo qubit, todo el sistema se vería afectado para cumplir la condición de normalización. Al alterar la amplitud de un estado, hay que actualizar el resto de amplitudes para que el sistema cumpla

$$\sum_{x=0}^{n} |\alpha_x|^2 = 1.$$

Recordemos que se podían ver las transformaciones como rotaciones de todo el espacio. Es decir, al operar con un solo qubit se actualizan 2^n amplitudes. Este es uno de los fenómenos por los que el poder de computo de los sistemas cuánticos es tan interesante.

3.2. Entrelazamiento

Anubody who is not shocked by quantum theory has not understood it.
-Niels Bohr

El entrelazamiento es un fenómeno cuántico que no tiene equivalente clásico. Cuando dos sistemas están entrelazados, se deben describir mediante un estado único que involucra a ambos sistemas, incluso estando separados espacialmente [12].

Decimos que dos sistemas pueden ser descritos individualmente

$$|v_1\rangle = \alpha_0 |0\rangle + \alpha_1 |1\rangle ,$$
$$|v_2\rangle = \beta_0 |0\rangle + \beta_1 |1\rangle ,$$

o como el producto tensorial entre ambos. El producto tensorial entre ellos es

$$|v_{12}\rangle = |v_1\rangle \otimes |v_2\rangle = \alpha_0\beta_0 |00\rangle + \alpha_0\beta_1 |01\rangle + \alpha_1\beta_0 |10\rangle + \alpha_1\beta_1 |11\rangle .$$

El estado $|v_{12}\rangle$, a su vez, puede ser factorizado en los dos estados anteriores. Este sería el caso para un estado arbitrario. Lo sorprendente es que a partir de dos estados $|v_1\rangle , |v_2\rangle$ individuales, se pueden crear estados conjuntos, $|v_{12}\rangle$, que no pueden ser factorizados, estos se dicen que son sistemas entrelazados.

3.2.1. Estados de Bell

Los estados de bell son los ejemplos más famosos de entrelazamiento, y pueden ser construidos con los operadores ya presentados. En concreto, para dos qubits se pueden crear usando la puerta M_{CNOT}, y la puerta Hadamard, que denotaremos como H.

Partiendo de dos qubits, q0 y q1, ambos en estado $|0\rangle$, aplicamos una Hadamard al primer qubit

$$H |q0\rangle = \frac{1}{\sqrt{2}} |0\rangle + \frac{1}{\sqrt{2}} |1\rangle ,$$
$$|q1\rangle = |0\rangle .$$

El estado del sistema compuesto, lo denotaremos con ψ, nos queda como

$$|\psi\rangle = \frac{1}{\sqrt{2}} |00\rangle + \frac{1}{\sqrt{2}} |10\rangle .$$

Si ahora aplicamos un operador M_{CNOT} entre ambos qubits con el primero como controlador, nos queda

$$M_{CNOT} |\psi\rangle = \frac{1}{\sqrt{2}} |00\rangle + \frac{1}{\sqrt{2}} |11\rangle .$$

Este se conoce como el estado de Bell $|\phi^+\rangle$. El resto los estados, llamados la base de Bell, se producen con el mismo circuito, pero cambiando los estados iniciales por el resto de estados en la base de bit del el espacio. Como en la evolución del sistemas, los ángulos se preservan, las transformaciones en una base ortogonal, producen otra base ortogonal. Entonces si tratamos el circuito anterior como un solo operador $U_b = M_{CNOT}(H \otimes I)$, el resto de estados de la base de Bell son:

$$U_b |01\rangle = \frac{1}{\sqrt{2}} |01\rangle + \frac{1}{\sqrt{2}} |10\rangle = |\psi^+\rangle$$
$$U_b |10\rangle = \frac{1}{\sqrt{2}} |00\rangle - \frac{1}{\sqrt{2}} |11\rangle = |\phi^-\rangle$$
$$U_b |11\rangle = \frac{1}{\sqrt{2}} |01\rangle - \frac{1}{\sqrt{2}} |10\rangle = |\psi^-\rangle$$

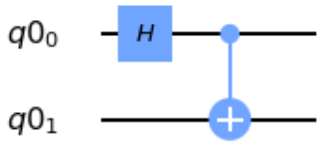


Figura 3.4: Circuito para crear un estado de Bell.

Trataremos de factorizar el estado $|\phi^+\rangle$:

Suponemos que existen dos sistemas independientes

$$|A\rangle = \alpha_0 |0\rangle + \alpha_1 |1\rangle\,,\ |B\rangle = \beta_0 |0\rangle + \beta_1 |1\rangle\,,$$

Entonces como sistema compuesto quedarían

$$|A\rangle \otimes |B\rangle = |AB\rangle = \alpha_0\beta_0 |00\rangle + \alpha_0\beta_1 |01\rangle + \alpha_1\beta_0 |10\rangle + \alpha_1\beta_1 |11\rangle\,.$$

Para que

$$|\phi^+\rangle = \frac{1}{\sqrt{2}} |00\rangle + \frac{1}{\sqrt{2}} |11\rangle\,,$$

pudiera ser descrito de esta forma, y por tanto factorizado se tiene que cumplir que

$$(\alpha_0\beta_0 = \alpha_1\beta_1 = \frac{1}{\sqrt{2}}) \ \wedge \ (\alpha_1\beta_0 = \alpha_0\beta_1 = 0)$$

lo cual es inconsistente, por tanto, un estado de Bell, no puede ser factorizado. La única forma de describirlo es como sistema compuesto.

Medida

La propiedad más asombrosa de estos estados, es que al medir uno de los qubits, al instante, sabemos también cual es el valor del otro qubit. Para

$$|\phi^+\rangle = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|10\rangle\,.$$

Si medimos el primer qubit con probabilidad de $\frac{1}{2}$ obtendremos el estado $|0\rangle$, y con la misma probabilidad obtendremos el estado $|1\rangle$. El sistema, quedará como un estado básico $|\psi\rangle$ en ambos casos. Por tanto obtendremos el estado de los dos qubits con una sola medida.

$$P[0] = \frac{1}{2} \;\Rightarrow\; |\psi\rangle = |00\rangle\,.$$

$$P[1] = \frac{1}{2} \;\Rightarrow\; |\psi\rangle = |11\rangle\,.$$

Esto sucede incluso si ambos qubits están separados espacialmente. El colapso de los estados durante la medida es una fuerza no local, no está limitada por la velocidad de la luz, ni depende de la distancia. Es decir, si tuviéramos dos qubits entrelazados, uno en la facultad de informática y otro en Plutón, al medir el qubit que está en la facultad, sabríamos el estado del qubit que está en Plutón. Este es el fenómeno que da lugar al teletransporte cuántico de información.

Paradoja de EPR

El hecho de que la mecánica cuántica fuera una teoría no local dio lugar a la *Paradoja de EPR (Einstein, Podolsky, Rosen)*. Esta paradoja dice que la mecánica cuántica no puede violar el principio de localidad. Albert Einstein en su artículo "Quanten-mechanik und wirklichkeit" [6] lo expresa así:

"...la siguiente idea caracteriza la independencia relativa de objetos que están muy alejados uno de otro en el espacio (A y B): una influencia externa en A no puede influir directamente sobre B; esto es conocido como el principio de acción local, y es empleado una y otra vez en teoría de campos. Si suprimiéramos por completo este axioma, resultaría inviable la idea de la existencia de sistemas semicerrados, y no podríamos postular leyes que se pudieran comprobar experimentalmente en el sentido aceptado.".

Más adelante John S. Bell, demostraría con sus *Desigualdades de Bell*, de manera experimental que este principio si podía ser violado.

3.3. Teletransporte y comunicación

Gracias a las propiedades no locales de los sistemas entrelazados, podemos aprovechar las correlaciones existentes entre ellos para operar entre ambos sin necesidad de comunicación física. En esta sección se explica cómo funciona el protocolo de teletransporte usando sistemas entrelazados.

3.3.1. Teorema de no clonación

Antes de hablar de teletransporte, primero tenemos que entender las limitaciones de comunicación que tienen los sistemas cuánticos.

Teorema 3.1 *No existe un circuito cuántico capaz de duplicar el valor de un estado cuántico arbitrario.*

Para demostrar el teorema, suponemos que es falso. Suponemos que existe un operador U_c que actúa sobre dos qubits, Q_0 y Q_1, con estados

$$|A\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\ ,$$

$$|0\rangle\,,$$

de tal manera, que

$$(\alpha\,|0\rangle + \beta\,|1\rangle) \otimes |0\rangle$$
$$\downarrow U_c$$
$$(\alpha\,|0\rangle + \beta\,|1\rangle) \otimes (\alpha\,|0\rangle + \beta\,|1\rangle).$$

Es decir, $U_c\,|A\rangle\,|0\rangle = |A\rangle\,|A\rangle\,.$

Siendo el estado $|A\rangle$, un estado arbitrario con cualquier $\alpha, \beta,$ si U_c funciona para cualquier valor, entonces funciona para para $\alpha = 1$,

$$U_c\,|00\rangle = |00\rangle\,.$$

Y para $\beta = 1$

$$U_c\,|10\rangle = |11\rangle\,.$$

Estos últimos estados forman la base del primer qubit. Sustituyendo, nos queda:

$$|A\rangle\,|0\rangle = (\alpha\,|0\rangle + \beta\,|1\rangle) \otimes |0\rangle = \alpha\,|00\rangle + \beta\,|10\rangle$$
$$\downarrow U_c$$
$$\alpha U_c\,|00\rangle + \beta U_c\,|10\rangle = \alpha\,|00\rangle + \beta\,|11\rangle$$

Por tanto llegamos a una contradicción, pues habíamos supuesto que obtendríamos dos qubits con el mismo estado para cualquier valor de α, β:

$$(\alpha\,|0\rangle + \beta\,|1\rangle) \otimes (\alpha\,|0\rangle + \beta\,|1\rangle) = \alpha^2\,|00\rangle + \alpha\beta\,|01\rangle + \alpha\beta\,|10\rangle + \beta^2\,|11\rangle\,.$$

El resultado al aplicar el supuesto operador es: $\alpha\,|00\rangle + \beta\,|11\rangle\,,$ pero este solo es el caso cuando $\alpha = 1 \vee \beta = 1$. Por tanto queda demostrado que no se puede realizar la clonación de un estado cuántico arbitrario.

3.3.2. Algoritmo de teletransporte

Suponemos que dos personas, Alice y Bob experimentan con protocolos de transporte cuánticos.

Dos qubits

La primera idea que se les ocurre, es usar un puerta M_{CNOT} para efectuar la comunicación. En este caso Alice dispone de un qubit Q_A en un estado arbitrario, y Bob dispone de un qubit Q_B en estado $|0\rangle$. Entonces aplican una puerta M_{CNOT} y obtienen un estado entrelazado (un estado de Bell para cualquier α, β).

$$\begin{gathered}(\alpha\,|0\rangle + \beta\,|1\rangle) \otimes |0\rangle = \alpha\,|00\rangle + \beta\,|10\rangle \\ \downarrow M_{CNOT} \\ \alpha\,|00\rangle + \beta\,|11\rangle\end{gathered}$$

Ahora Alice, mide su qubit, pero en este caso efectuará la medida en la base de signo. Haciendo un cambio de base para Q_A, el sistema nos queda:

$$\alpha(\frac{1}{\sqrt{2}}\,|+\rangle + \frac{1}{\sqrt{2}}\,|-\rangle)\,|0\rangle + \beta(\frac{1}{\sqrt{2}}\,|+\rangle - \frac{1}{\sqrt{2}}\,|-\rangle)\,|1\rangle =$$

$$\frac{1}{\sqrt{2}}\,|+\rangle\,(\alpha\,|0\rangle + \beta\,|1\rangle) + \frac{1}{\sqrt{2}}\,|-\rangle\,(\alpha\,|0\rangle - \beta\,|1\rangle)$$

Si al realizar la medida, Alice obtiene el estado $|+\rangle$, el sistema queda:

$$|+\rangle\,(\alpha\,|0\rangle + \beta\,|1\rangle).$$

Es decir, $|Q_B\rangle = (\alpha\,|0\rangle + \beta\,|1\rangle)$, tiene el mismo estado que Q_A al iniciar el experimento. Y en el caso de que al realizar la medida, Alice obtenga el estado $|-\rangle$, el sistema queda:

$$|-\rangle\,(\alpha\,|0\rangle - \beta\,|1\rangle).$$

Entonces aplicamos una puerta Z a Q_B

$$Z(\alpha\,|0\rangle - \beta\,|1\rangle) = (\alpha\,|0\rangle + \beta\,|1\rangle),$$

y obtenemos el mismo resultado. En ambos casos hemos teletransportado el estado de un qubit al otro. Pero este caso, tiene que existir un canal de comunicación convencional entre ambos qubits para preparar el experimento con la puerta M_{CNOT}.

Tres qubits

Después del primer experimento, se les ocurre intentar el teletransporte con tres qubits. En este caso Alice y Bob, crean un estado de Bell $|\phi^+\rangle$ en su laboratorio de la universidad de Granada, y tras crearlo Bob coge su qubit y se marcha a un laboratorio en Plutón (¡Recordemos que el entrelazamiento sigue funcionando en Plutón!). Una vez ambas están en sus respectivos laboratorios a distancia suficiente para que el experimento sea impresionante, Alice añade al experimento otro qubit, cuyo contenido será teletransportado. Quedan tres qubits Q_{A0}, Q_{A1}, con Alice, y Q_B con Bob.

Q_{A0}	Q_{A1}, Q_B
$\alpha\,\|0\rangle + \beta\,\|1\rangle$	$\frac{1}{\sqrt{2}}\,\|00\rangle + \frac{1}{\sqrt{2}}\,\|11\rangle$

El sistema nos quedaría como

$$(\alpha\,|0\rangle + \beta\,|1\rangle) \otimes (\frac{1}{\sqrt{2}}\,|00\rangle + \frac{1}{\sqrt{2}}\,|11\rangle) =$$

$$\frac{\alpha}{\sqrt{2}}\,|000\rangle + \frac{\alpha}{\sqrt{2}}\,|011\rangle + \frac{\beta}{\sqrt{2}}\,|100\rangle + \frac{\beta}{\sqrt{2}}\,|111\rangle\,.$$

Ahora Alice, aplica una puerta M_{CNOT} en su laboratorio entre Q_{A0} y Q_{A1}

$$\frac{\alpha}{\sqrt{2}}\,|000\rangle + \frac{\alpha}{\sqrt{2}}\,|011\rangle + \frac{\beta}{\sqrt{2}}\,|100\rangle + \frac{\beta}{\sqrt{2}}\,|111\rangle\,.$$

$$\downarrow M_{CNOT}(Q_{A0}, Q_{A1})$$

$$\frac{\alpha}{\sqrt{2}}\,|000\rangle + \frac{\beta}{\sqrt{2}}\,|101\rangle + \frac{\alpha}{\sqrt{2}}\,|011\rangle + \frac{\beta}{\sqrt{2}}\,|110\rangle\,.$$

Tras aplicar el operador, mide el segundo qubit Q_{A1}, nos quedan dos casos:

Si obtiene $|0\rangle$, el sistema queda para los otros qubits

$$\alpha\,|00\rangle + \beta\,|11\rangle\,.$$

Entonces nos encontramos en mismo caso que el primer experimento y sabemos que si Alice mide Q_{A0} en las bases de signo, conseguiremos teletransportar el estado de Q_{A0} a Q_B.

En el caso de que obtenga $|1\rangle$, el sistema quedará para los otros qubits

$$\alpha\,|01\rangle + \beta\,|10\rangle\,,$$

pero aplicando una puerta X a Q_B

$$I \otimes Z(\alpha\,|01\rangle + \beta\,|10\rangle) = \alpha\,|00\rangle + \beta\,|11\rangle\,,$$

volvemos a la situación anterior y también se puede efectuar el teletransporte de información entre Q_{A0} y Q_B. En este caso, hemos conseguido teletransportar un estado entre qubits.

En el siguiente diagrama se muestra el circuito de teletransporte. En este circuito, en vez de medir en la base de signo, se aplica una puerta Hadamard, esto es equivalente a medir en las bases de signo. Se debe, a que la transformación Hadamard cambia las amplitudes del sistema de la misma forma que un cambio de base de bit a signo. Por tanto, aplicar una puerta Hadamard y luego medir en la base de bit, es lo mismo que medir en la base de signo. C_0 y C_1 son registros clásicos donde se guarda la información obtenida durante la medida.

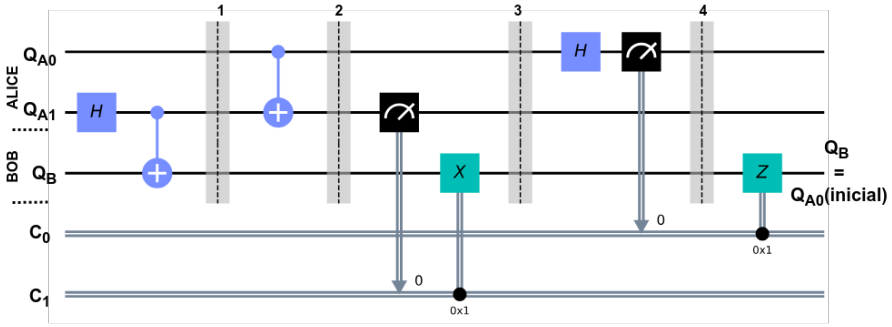


Figura 3.5: Circuito de teletransporte.

1. Se prepara un estado de Bell entre Q_{A1} y Q_B.
2. Se aplica una puerta M_{CNOT} entre Q_{A0} y Q_{A1}.
3. Medimos Q_{A1}. Si obtenemos $|1\rangle$, aplicamos una puerta X a Q_B.
4. Aplicamos una puerta H a Q_{A0}, y medimos. Si obtenemos $|1\rangle$, aplicamos una puerta Z a Q_B.

En resumen, preparando un estado de Bell, se puede transmitir un estado cuántico entre dos qubits que no están en contacto entre sí. El estado inicial que queremos teletransportar, es destruido en el primer qubit para que pueda llegar al segundo qubit, por tanto no se viola el teorema de no clonación. Cabe destacar, que para realizar el teletransporte de un estado cuántico, hace falta un canal de comunicación clásico entre Alice Y Bob para enviar los resultado de las medidas.

3.3.3. Codificación superdensa

En el algoritmo de teletransporte, se consigue teletransportar un estado cuántico con ayuda de un canal de comunicación clásico. El protocolo de codificación superdensa, nos permite transmitir dos bis clásicos, usando un qubit como contenedor. Se denomina superdensa, porque se codifican dos bits con un solo qubit. Además, se trata de un canal de comunicación completamente seguro porque se utiliza el entrelazamiento entre qubits como método de transmisión.

En este caso, Alice y Bob comparten un estado de bell $Q_{AB} = |\phi^+\rangle$ entre Q_A y Q_B, y Alice quiere enviar información clásica a Bob. Alice puede enviar dos bits, y dependiendo de cuales sean, aplicará diferentes transformaciones a Q_A. Se muestran en la siguiente tabla:

Bits	Operador	Estado Q_{AB}
00	I	$\frac{1}{\sqrt{2}}(\|00\rangle + \|11\rangle)$
01	X	$\frac{1}{\sqrt{2}}(\|01\rangle + \|10\rangle)$
10	Z	$\frac{1}{\sqrt{2}}(\|00\rangle - \|11\rangle)$
11	XZ	$\frac{1}{\sqrt{2}}(\|01\rangle - \|10\rangle)$

Cada nuevo estado, corresponde con un estado de la base de Bell. Tras realizar las operaciones, Alice le envía su qubit Q_A, a Bob. Ahora, Bob tiene ambos qubits, es decir Q_{AB}. Para decodificar la información que Alice le ha envidado, aplicará el circuito necesario para obtener un estado de Bell invirtiendo el orden de las operaciones. Primero aplicar M_{CNOT} con Q_A como controlador, y luego H sobre Q_A. Se muestran los resultados para cada caso en la siguiente tabla:

Q_{AB}	$M_{CNOT}Q_{AB}$	$(H \otimes I)M_{CNOT}Q_{AB}$
$\frac{1}{\sqrt{2}}(\|00\rangle + \|11\rangle)$	$\frac{1}{\sqrt{2}}(\|00\rangle + \|10\rangle)$	$\|00\rangle$
$\frac{1}{\sqrt{2}}(\|01\rangle + \|10\rangle)$	$\frac{1}{\sqrt{2}}(\|01\rangle + \|11\rangle)$	$\|01\rangle$
$\frac{1}{\sqrt{2}}(\|00\rangle - \|11\rangle)$	$\frac{1}{\sqrt{2}}(\|00\rangle - \|10\rangle)$	$\|10\rangle$
$\frac{1}{\sqrt{2}}(\|01\rangle - \|10\rangle)$	$\frac{1}{\sqrt{2}}(\|01\rangle - \|11\rangle)$	$\|11\rangle$

Bob siempre obtiene los bits que Alice le ha enviado.

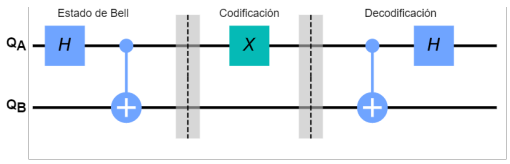


Figura 3.6: Circuito del protocolo de codificación superdensa para '01'.

3.4. Circuitos clásicos

En está se sección se expone como implementar circuitos clásicos en sistemas cuánticos. Gracias a ello, podremos implementar cualquier función computable por un ordenador clásico, en un ordenador cuántico [19].

3.4.1. Computación reversible

Los sistemas cuánticos presentan una evolución determinística y sin perdida de información, esto se debe a la naturaleza reversible de los operadores que actúan en el sistema. Los circuitos clásicos son implementados por operadores generalmente no reversibles, se pierde información del sistema. Por ejemplo, una puerta 'NOT', es un operador reversible, pero una puerta 'AND', no es reversible. Una puerta 'AND', para dos entradas genera una sola salida, y la información inicial del sistema, no se puede recuperar. Para que un operador pueda ser reversible, tiene que implementar un función biyectiva. Como los sistemas cuánticos presentan una evolución reversible, tenemos que presentar los circuitos clásicos de tal forma que sean reversibles.

Toffoli

La puerta de Toffoli o CCNOT, es una puerta reversible, que actúa sobre tres bits, y es capaz de implementar las puertas NOT, AND, NAND y la operación de copia. Se describe como

$$T : B^3 \to B^3 : T(x_1, x_2, x_3) = (x_1, x_2, (x_1 \wedge x_2) \oplus x_3).$$

El resultado de esta puerta, queda guardado en el bit x_3.
Si fijamos $x_3 = 0$, obtenemos una puerta AND

$$T : B^3 \to B^3 : T(x_1, x_2, 0) = (x_1, x_2, x_1 \wedge x_2).$$

Si fijamos $x1 = x_2 = 0$, obtenemos una puerta OR

$$T : B^3 \to B^3 : T(0, 0, x_3) = (0, 0, \neg x_3).$$

Si fijamos $x_3 = 1$, obtenemos una puerta NAND

$$T : B^3 \to B^3 : T(x_1, x_2, 1) = (x_1, x_2, \neg(x_1 \wedge x_2)).$$

Para copiar el bit x_2, establecemos $x_1 = 1$, y $x_3 = 0$

$$T : B^3 \to B^3 : T(1, x_2, 0) = (1, x_2, x_2).$$

Con estas operaciones, la puerta de Toffoli es capaz de implementar cualquier circuito clásico, por tanto es universal.

3.4.2. Implementando circuitos clásicos

La puerta de Toffoli puede actuar sobre sistemas cuánticos como un operador unitario. Puede ser implementada en circuito cuántico mediante operadores H, M_{CNOT}, T y S.

T y S son operadores de rotación para ángulos $\Phi_T = \frac{i\pi}{4}$ y $\Phi_S = \frac{i\pi}{2}$.

$$T = \begin{pmatrix} 1 & 0 \\ 0 & e^{\frac{i\pi}{4}} \end{pmatrix}$$

$$S = \begin{pmatrix} 1 & 0 \\ 0 & e^{\frac{i\pi}{2}} \end{pmatrix}$$

Las siguientes figuras, ilustran el circuito que implementa el operador y su matriz.

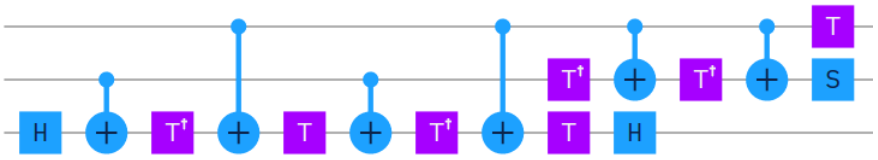


Figura 3.7: Implementación con M_{CNOT}, H, T y S.

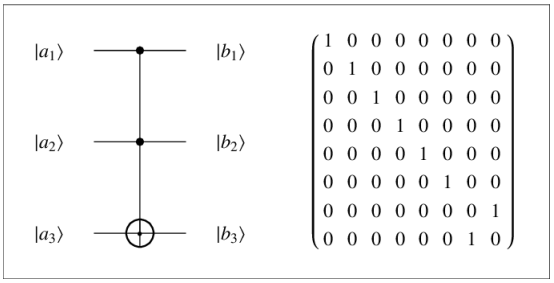


Figura 3.8: Circuito y matriz unitaria del operador de Toffoli.

Ahora sabemos, que la puerta de Toffoli puede ser implementada con un circuito cuántico. Dado cualquier circuito clásico C, podemos sustituir las puertas lógicas que lo constituyen por Toffoli, e implementarlo como un circuito cuántico R_C. Al aplicar este operador a un sistema cuántico que codifique la entrada x para C, el resultado será una superposición entre $C(x)$ y bits redundantes. Los bits redundantes, aparecen dado que la puerta Toffoli computa la salida en x_3 y el resto de bits son para poder hacer la operación reversible, pero no aportan información sobre el resultado de la función. Al igual pasa al codificar la entrada, no todos los bits codifican x,

también hay bits de control. El circuito nos quedaría como se muestra en la siguiente figura.

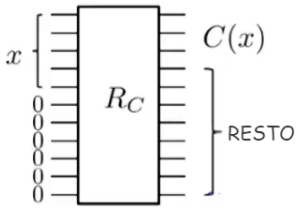


Figura 3.9: Operador R_C.

El problema de esta superposición, es que puede presentar interferencias entre $C(x)$ y 'resto'. Para obtener el resultado, se hace una copia con M_{CNOT} de $C(x)$ en otro registro, y se revierte la operación con R_C^*.

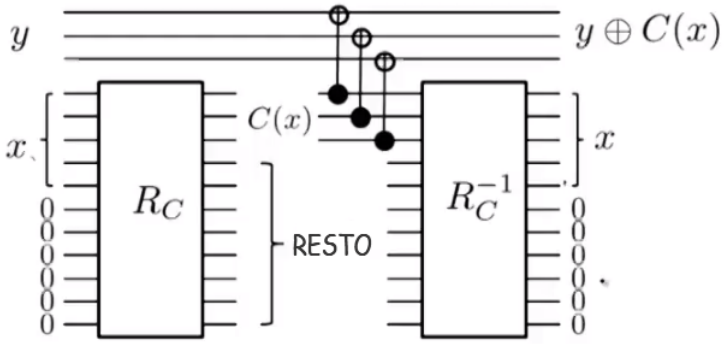


Figura 3.10: Circuito U_C.

De esta forma podemos implementar cualquier función computable clásicamente en un sistema cuántico. Podemos denotar el circuito como U_C, que actúa de la siguiente manera:

$$U_C \left|x\right\rangle \left|y\right\rangle \left|0\right\rangle^{\otimes n} = \left|x\right\rangle \left|C(x) \oplus y\right\rangle \left|0\right\rangle^{\otimes n}.$$

Se puede obviar $\left|0\right\rangle^{\otimes n}$ en la notación, y nos queda

$$U_C \left|x\right\rangle \left|y\right\rangle = \left|x\right\rangle \left|C(x) \oplus y\right\rangle.$$

3.5. Algoritmo de Búsqueda

En esta sección se expone el Algoritmo de búsqueda de Grover (creador del algoritmo en 1996), que permite realizar una búsqueda dentro de una secuencia no ordenada de datos en tiempo $O(\sqrt{N})$ [5] [20].

3.5.1. Introducción

La búsqueda puede ser representada por un función $f : \{0, 2^n - 1\} \rightarrow \{0, 1\}$, donde 2^n representa los índices de los objetos a buscar, y $f(x) = 1$, solo cuando x es solución al problema de búsqueda. Las soluciones al problema se denotan como x^* y el número de soluciones es $1 \leq S \leq 2^n$.

$$f(x) = \begin{cases} 1 & \text{si x=x}^* \\ 0 & \text{en otro caso,} \end{cases}$$

El algoritmo, se basa en la aplicación de dos operadores, Θ (Oráculo) y D (Operador de difusión de Grover).

El Oráculo (Θ), implementa el circuito clásico que computa f, y determina que estados representan una solución. Para determinar las soluciones, invierte la amplitud de su estado.

$$\Theta(|x\rangle) = \begin{cases} \text{- } |x\rangle & \text{si f(x)=1} \\ |x\rangle & \text{en otro caso,} \end{cases}$$

Tras aplicar Θ, se aplica el operador de difusión de Grover (D). Este, para cada estado de la superposición, realiza una inversión de su amplitud sobre la media de todas las amplitudes(μ).

$$\sum_{i=0}^{i=2^n} \alpha_i |x_i\rangle \xrightarrow{D} \sum_{i=0}^{i=2^n} (2\mu - \alpha_i) |x_i\rangle$$

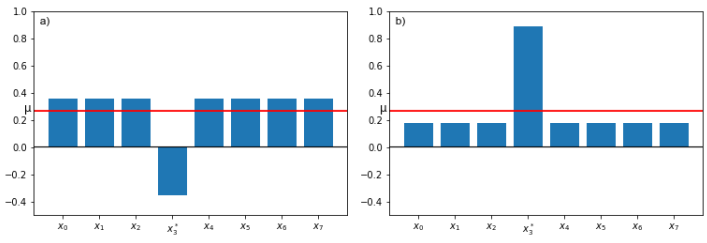


Figura 3.11: Efecto en las amplitudes de un sistema tras aplicar Θ (a), y $D\Theta$ (b) para una superposición de n=3.

La composición de ambos operadores, conforma una amplificación de las amplitudes correspondientes a los estados que representan una solución de f. Esta composición se denota con el operador G. El algoritmo aplica G reiteradamente para maximizar la amplitud de los estados solución.

3.5.2. Operador de amplificación

Oráculo

Dada la función f, Θ debe averiguar que valores de entrada son solución e invertir sus amplitudes. Para ello, implementará el circuito clásico que computa f, como se ha expuesto en apartados anteriores:

$$U_f \left|x\right\rangle \left|y\right\rangle = \left|x\right\rangle \left|f(x) \oplus y\right\rangle .$$

La entrada, $|x\rangle$ de f, será una superposición equiprobable de $2^n = N$ estados correspondientes a los índices de búsqueda. Esta superposición, se consigue aplicando $H^{\otimes n}$ a una entrada $|0^{\otimes n}\rangle$. Gracias a esta superposición, se computan todas las posibles entradas de la función. Para cambiar la amplitud de los estados solución, el Qubit auxiliar 'y', será sustituido por $|1\rangle$ y le aplicaremos H también. Nos queda como entrada:

$$H^{\otimes n+1} \left|0^{\otimes n}\right\rangle \left|1\right\rangle = \frac{1}{\sqrt{2^n}} \sum_{i=0}^{i=2^n} \left|x_i\right\rangle \otimes \left|-\right\rangle = \left|x\right\rangle \left|-\right\rangle$$

Si para esta entrada, aplicamos U_f, queda

$$U_f \left|x\right\rangle \left|-\right\rangle = \left|x\right\rangle \left|f(x) \oplus \left|-\right\rangle\right\rangle .$$

Para $(f(x) \oplus |-\rangle)$, podemos observar dos casos posibles:

- $f(x) = 0$, entonces $f(x) \oplus |-\rangle = f(x) \oplus \frac{|0\rangle - |1\rangle}{\sqrt{2}} = \frac{|0\rangle - |1\rangle}{\sqrt{2}} = |-\rangle$.
- $f(x) = 1$, entonces $f(x) \oplus |-\rangle = f(x) \oplus \frac{|0\rangle - |1\rangle}{\sqrt{2}} = \frac{|1\rangle - |0\rangle}{\sqrt{2}} = -|-\rangle$.

Se produce una inversión de la amplitud cuando x es solución, es decir $x = x^*$. Podemos expresar $(f(x) \oplus |-\rangle)$ de la siguiente manera

$$f(x) \oplus \left|-\right\rangle = (-1)^{f(x)} \left|-\right\rangle .$$

Podemos reformular como

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

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

$$= \frac{1}{\sqrt{2^n + 1}} \sum_{i=0}^{i=2^n} \left|x_i\right\rangle (-1)^{f(x_i)} \left|-\right\rangle .$$

Finalmente, la expresión que nos queda es

$$\frac{1}{\sqrt{2^n+1}}\sum_{i=0}^{i=2^n}(-1)^{f(x_i)}\left|x_i\right\rangle\left|-\right\rangle.$$

Este estado, es una superposición equiprobable de todas las entradas x de f, pero con la amplitud de las entradas solución, x^*, invertidas.

Resumiendo, nos queda el efecto de Θ, como

$$\Theta\left|0^{\otimes n}\right\rangle\left|1\right\rangle = U_f H^{\otimes n+1}\left|0^{\otimes n}\right\rangle\left|1\right\rangle = \frac{1}{\sqrt{2^n+1}}\sum_{i=0}^{i=2^n}(-1)^{f(x_i)}\left|x_i\right\rangle\left|-\right\rangle.$$

El Qubit auxiliar se suele ignorar y queda simplemente

$$\frac{1}{\sqrt{2^n+1}}\sum_{i=0}^{i=2^n}(-1)^{f(x_i)}\left|x_i\right\rangle.$$

Operador de difusión

El operador de difusión de Grover, realiza una inversión de todas las amplitudes de un estado en superposición, sobre la media(μ) de estas amplitudes. Para ver como construir este operador, tenemos que examinar como computar μ. Para ello, usaremos el operador de difusión de un estado $|x\rangle$ en una superposición equiprobable.

$$\left|x\right\rangle = \frac{1}{\sqrt{2^n}}\sum_{i=0}^{i=2^n}\left|x_i\right\rangle$$

$$P_x = \left|x\right\rangle\left\langle x\right| = \begin{pmatrix} \frac{1}{2^n} & \frac{1}{2^n} & \cdots & \frac{1}{2^n} \\ \frac{1}{2^n} & \frac{1}{2^n} & \cdots & \frac{1}{2^n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{2^n} & \frac{1}{2^n} & \cdots & \frac{1}{2^n} \end{pmatrix}$$

Al aplicar P_x sobre $|\psi\rangle$, se computa la media de todas las amplitudes de $|\psi\rangle$ en cada una de las posiciones del vector resultante.

$$\begin{pmatrix} \frac{1}{2^n} & \frac{1}{2^n} & \cdots & \frac{1}{2^n} \\ \frac{1}{2^n} & \frac{1}{2^n} & \cdots & \frac{1}{2^n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{2^n} & \frac{1}{2^n} & \cdots & \frac{1}{2^n} \end{pmatrix}\begin{pmatrix} \psi_1 \\ \psi_2 \\ \vdots \\ \psi_{2^n} \end{pmatrix} = \begin{pmatrix} \frac{\sum_{i=1}^{i=2^n}\psi_i}{2^n} \\ \frac{\sum_{i=1}^{i=2^n}\psi_i}{2^n} \\ \vdots \\ \frac{\sum_{i=1}^{i=2^n}\psi_i}{2^n} \end{pmatrix}$$

Es decir, la matriz de proyección P de un estado en superposición equiprobable $|x\rangle$, aplicada a un estado $|\psi\rangle$, calcula la media de las amplitudes

de $|\psi\rangle$. Para conseguir esta matriz, se aplican operadores Hadamard a la matriz de proyección $P_0 = |0\rangle\langle 0|$

$$H|0\rangle\langle 0|H = |H0\rangle\langle H^*0| = |H0\rangle\langle H0| = |x\rangle\langle x|$$

Ahora podemos construir el operador D como

$$D = 2P_x - I$$

Para una entrada ψ_i de $|\psi\rangle$,

$$\alpha_i \xrightarrow{D} (2\mu - \alpha_i),$$

y para $|\psi\rangle$

$$\sum_{i=0}^{i=2^n} \alpha_i |\psi_i\rangle \xrightarrow{D} \sum_{i=0}^{i=2^n} (2\mu - \alpha_i)|\psi_i\rangle .$$

Por tanto, $D = 2P_x - I$ realiza una inversión sobre la media de las amplitudes de un estado.

Circuito

El circuito, G, que implementa estos operadores resulta como se indica en la siguiente figura:

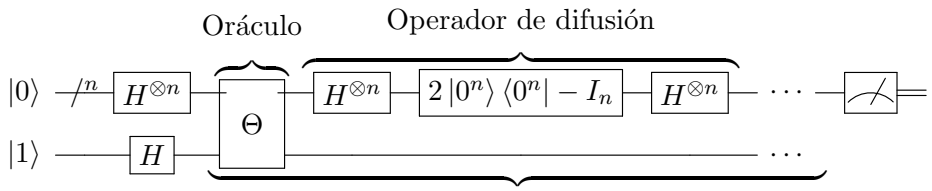


Figura 3.12: Operador de Amplificación G.

Cada iteración sobre G supone una amplificación de los estados solución, cuantas más veces apliquemos G, más probabilidades hay de medir una solución. La siguiente gráfica, muestra la probabilidad de medir un estado solución tras una aplicación de G.

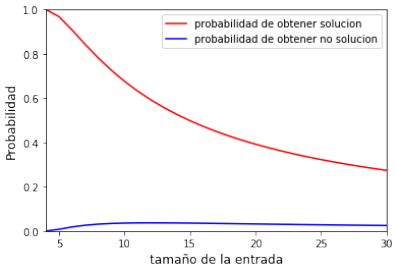


Figura 3.13: Probabilidades del sistema tras una aplicación de G.

3.5.3. Interpretación geométrica

En esta sección se presenta un análisis del procedimiento de amplificación, centrandose en el peor de los casos, cuando el número de soluciones es uno. El estado inicial del sistema es $|x\rangle = |\psi^{(0)}\rangle$, antes de aplicar G. Podemos definir el estado después de la r-ésima aplicación de G como

$$|\psi^{(r)}\rangle = G\,|\psi^{r-1}\rangle = G^{r}\,|x\rangle = (D\Theta)^{r}\,|x\rangle\,.$$

Para realizar la interpretación geométrica, vamos a construir dos estados ortogonales a partir de $|x\rangle$:

El primero es el estado que representa la solución

$$|s\rangle = |x^{*}\rangle\,.$$

El segundo, es el subespacio de $|x\rangle$ compuesto por el conjunto de estados no solución

$$|u\rangle = \frac{1}{\sqrt{2^{n}-1}} \sum_{x \neq x^{*}} |x\rangle\,.$$

$|u\rangle$ es $|x\rangle$, pero restando la componente $|s\rangle$ y normalizado. Estos estados, forman una base ortonormal en un subespacio bidimensional de $|x\rangle$. Por tanto, $|x\rangle$, se puede expresar como una combinación lineal de $|u\rangle$ y $|s\rangle$

$$|x\rangle = a\,|u\rangle + b\,|s\rangle\,,\ \forall a, b \in R.$$

Los operadores Θ y D, también son transformaciones que relacionan vectores dentro del subespacio

$$\Theta\,|x\rangle = a\Theta\,|u\rangle + b\Theta\,|s\rangle = a\,|u\rangle - b\,|s\rangle\,,$$

$$D\,|x\rangle = (2\,|x\rangle\,\langle x| - I)\,|x\rangle = 2\,\langle x|\,|x\rangle\,|x\rangle - |x\rangle = \lambda\,|x\rangle - |x\rangle\,.$$

Por tanto, podemos interpretar el algoritmo en este subespacio, que es isomorfo con el plano.

Reflexión en el plano

Los operadores Θ y D son reflexiones en el plano. Para Θ, se ve claramente que es una reflexión sobre $|u\rangle$, $\Theta\,|x\rangle = a\,|u\rangle - b\,|s\rangle$. Para entender D, examinaremos primero una reflexión general de un vector $|k\rangle$, sobre otro $|v\rangle$.

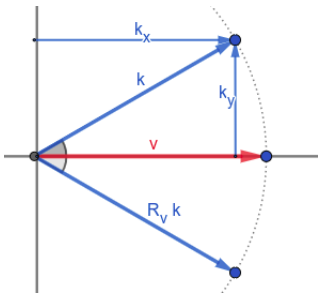


Figura 3.14: Reflexión de k sobre v.

Siendo R_v, el operador de reflexión de $|k\rangle$ sobre $|v\rangle$, y $|k_x\rangle, |k_y\rangle$ las componentes de $|k\rangle$ en los distintos ejes del plano:

$$R_v |k\rangle = |k\rangle - 2 |k_y\rangle ;$$

$$|k_x\rangle = \lambda |v\rangle ; \langle v| |k_x\rangle = 0;$$

$$\langle v| |k\rangle = \langle v| |k_x\rangle + \langle v| |k_x\rangle = \langle v| |k_x\rangle = \lambda \langle v| |v\rangle = \lambda.$$

$$|k_x\rangle = \langle v| |k\rangle |v\rangle ;$$

$$|k_y\rangle = |k\rangle - |k_x\rangle = |k\rangle - \langle v| |k\rangle |v\rangle .$$

$$R_v |k\rangle = |k\rangle - 2(|k\rangle - \langle v| |k\rangle |v\rangle) = 2 |v\rangle \langle v| |k\rangle - |k\rangle$$

$$R_v |k\rangle = (2 |v\rangle \langle v| \ - I) |k\rangle$$

Por tanto, el operador de reflexión sobre un vector arbitrario $|v\rangle$ es

$$R_v = 2 |v\rangle \langle v| \ - I.$$

Como sabemos que $D = 2 |x\rangle \langle x| - I$, este conforma una reflexión sobre $|x\rangle$ en el plano. Es decir, $D = R_x$ y $\Theta = R_u$.

Siendo α el ángulo entre $|x\rangle$ y $|u\rangle$, podemos ver que tras la aplicación de Θ, el ángulo entre $\Theta |x\rangle$ y $|u\rangle$, es $(-\alpha)$. Al aplicar D, $\Theta |x\rangle$ se refleja sobre $|x\rangle$ y el ángulo entre $D\Theta |x\rangle$ y $|u\rangle$ queda como $(3 \cdot \alpha)$.

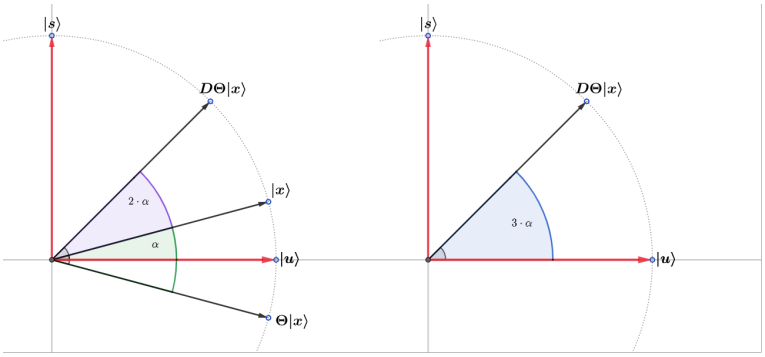


Figura 3.15: Aplicación de G sobre $|x\rangle$ visualizado en el plano.

Entonces, tras la primera aplicación de G, el ángulo queda como $\delta^{(1)} = (3 \cdot \alpha)$. Este, es el ángulo entre $|\psi^{(1)}\rangle$ (el estado del sistema tras aplicar $G^{(1)}$) y $|u\rangle$ ("estado no solución"). Para $|\psi^{(0)}\rangle$, $\delta^{(0)} = \alpha$, y generalmente, para $|\psi^{(r)}\rangle$,

$$\delta^{(r)} = \delta^{(r-1)} + 2 \cdot \alpha,$$

$$\delta^{(0)} = \alpha,\ \delta^{(1)} = \alpha + 2 \cdot \alpha,\ ...,\ \delta^{(r)} = (2r+1)\alpha.$$

Podemos expresar el estado $|\psi^{(r)}\rangle$, como

$$|\psi^{(r)}\rangle = \cos\delta^{(r)}\,|u\rangle + \sin\delta^{(r)}\,|s\rangle\,.$$

Y la probabilidad de obtener la solución $|s\rangle = |x^*\rangle$ es

$$P[|s\rangle] = sin^2\delta^{(r)}.$$

Por tanto, para obtener el estado que representa la solución al problema de búsqueda, hay que encontrar el valor de r que maximice la probabilidad de obtener $|s\rangle$. Para $\delta^{(r)} = \frac{\pi}{2}$, $P[|s\rangle] = 1$, hay que encontrar r de manera que

$$\delta^{(r)} = \frac{\pi}{2} = (2r+1)\alpha$$

Podemos obtener α, con la proyección de $|x\rangle$ sobre $|u\rangle$

$$\cos\alpha = \langle x|\,|u\rangle = \sum_{i=0}^{i=2^n} x_i \cdot u_i = (2^n - 1)\left(\frac{1}{\sqrt{2^n-1}}\right)\left(\frac{1}{\sqrt{2^n}}\right) = \sqrt{\frac{2^n-1}{2^n}}.$$

Como $[\cos^2\alpha + \sin^2\alpha = 1]$, podemos despejar y obtener

$$\sin\alpha = \sqrt{\frac{1}{2^n}}.$$

El algoritmo se analiza para valores altos de n, entonces, el valor de $\sin\alpha$ será muy pequeño, por tanto, se puede usar la *aproximación del ángulo pequeño*

$$\sin\alpha = \alpha = \sqrt{\frac{1}{2^n}}.$$

Nos queda

$$\delta^{(r)} = \frac{\pi}{2} = (2r+1)\left(\sqrt{\frac{1}{2^n}}\right).$$

Despejando r, obtenemos la siguiente expresión

$$r = \frac{\sqrt{2^n}\ \frac{\pi}{2} - 1}{2}.$$

Como n tomará valores altos, el (-1) que forma parte del dividendo, es despreciable para el análisis del algoritmo.

$$r = \frac{\pi}{4}\sqrt{2^n}.$$

Finalmente, obtenemos el número de iteraciones necesarias para obtener la solución del problema. También comprobamos que la complejidad temporal del algoritmo, en el peor de los casos, es $O(\sqrt{N}) = O(\sqrt{2^n})$. Esto supone una mejora cuadrática respecto a la versión clásica del algoritmo de búsqueda para una secuencia de datos no ordenada $O(2^n)$.

Parte III

RESOLUCIÓN EXPERIMENTAL DEL TRABAJO

Capítulo 4

Recursos utilizados

En este capítulo se presentan los recursos utilizados para la parte experimental de este trabajo. Se usa como marco de desarrollo la herramienta *IBM Quantum Experience*, que da acceso a ordenadores cuánticos reales y que ofrece el lenguaje *Qiskit* para programarlos [1].

4.1. Características de los computadores cuánticos utilizados

En este apéndice se muestran las características de los ordenadores cuánticos usados en este trabajo, estos dispositivos reciben el nombre de las ciudades de *Burlington, London y Melborune* . Previamente se definen algunos conceptos necesarios.

Decoherencia cuántica

La decoherencia, es el proceso de deconstrucción de la interferencia cuántica, cuando un estado en superposición colapsa a un estado puro. La interacción del sistema cuántico con el ambiente es la causa del proceso de decoherencia más aceptada. En condiciones ideales, se trabajaría con sistemas cerrados que no tienen contacto con el exterior, pero en la practica es difícil aislar totalmente los sistemas. Cuanto menor sea el tiempo de decoherencia de un sistema, más difícil será realizar experimentos sin que los resultados ideales sean perturbados.

En las características de los dispositivos se proporcionan dos tiempos de decoherencia: T1 y T2. Estos dos tiempos miden los efectos de la decoherencia para bases ortonormales de un sistema. T1 se refiere al tiempo de relajación, y corresponde con el tiempo que tarda en decaer un qubit preparado en un estado $|1\rangle$ a un estado $|0\rangle$. Mientras que T2, se refiere al tiempo de desfase, y corresponde con el tiempo que tarda en decaer un qubit preparado en un estado $|+\rangle$ a un estado $|-\rangle$.

Puertas atómicas

Los dispositivos implementan operaciones atómicas o físicas, con las que construyen todas las posibles operaciones. Estas son la puerta CNOT y las puertas U1, U2 y U3.

La puerta U3 es una generalización parametrizada de cualquier puerta de un solo qubit.

$$U_3(\theta, \phi, \lambda) = \begin{bmatrix} \cos(\theta/2) & -e^{i\lambda}\sin(\theta/2) \\ e^{i\phi}\sin(\theta/2) & e^{i\lambda+i\phi}\cos(\theta/2) \end{bmatrix}$$

Y las puertas U1 y U2, son simplemente casos específicos de ella:

$$U_3(\tfrac{\pi}{2}, \phi, \lambda) = U_2 = \tfrac{1}{\sqrt{2}} \begin{bmatrix} 1 & -e^{i\lambda} \\ e^{i\phi} & e^{i\lambda+i\phi} \end{bmatrix} \qquad U_3(0, 0, \lambda) = U_1 = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\lambda} \end{bmatrix}$$

Antes de ejecutar un circuito, todas la puertas definidas son traducidas a CNOT, U1, U2 y U3.

Volumen cuántico

El volumen cuántico es una medida diseñada para encapsular el rendimiento de un ordenador cuántico. Describir los detalles de este proceso quedan fuera del alcance de este trabajo, pero a grandes rasgos, este índice, se obtiene mediante un protocolo que mide el rendimiento del dispositivo al aplicar en paralelo puertas cuánticas de dos qubits con parámetros aleatorios, sobre un subconjunto de qubits del sistema. Actualmente, el ordenador con mayor volumen cuántico con el que cuenta IBM, es su dispositivo 'Montreal', con un volumen de 64.

Ordenadores cuánticos

ibmq_burlington

- **Número de qubits:** 5.
- **Volumen cuántico** 8.
- **Número máximo de ejecuciones** 8192.
- **Disponible desde:** 13-9-2019.

	T1	T2	Error puerta U1	Error puerta U2	Error puerta U2
Q0	102.82728 us	76.00597 us	0	0.00038	0.00076
Q1	77.4008 us	95.55764 us	0	0.00074	0.00148
Q2	53.08205 us	89.10213 us	0	0.00054	0.00108
Q3	90.33801 us	87.14363 us	0	0.00044	0.00088
Q4	78.86252 us	19.42975 us	0	0.00041	0.00081

La siguiente imagen representa el mapa topológico del dispositivo, los ratios de error para las puertas CNOT, H, y el ratio de error de lectura.

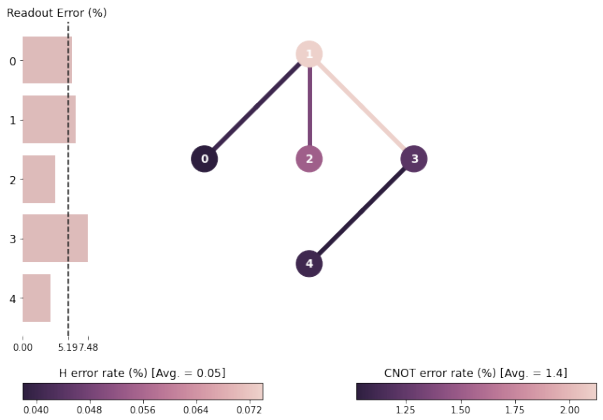


Figura 4.1: Mapa topológico del ordenador ibmq_burlington.

ibmq_london

- **Número de qubits:** 5.
- **Volumen cuántico** 16.
- **Número máximo de ejecuciones** 8192.
- **Disponible desde:** 13-9-2019.

	T1	T2	Error puerta U1	Error puerta U2	Error puerta U2
Q0	60.66452 us	99.10984 us	0	0.00037	0.00074
Q1	68.02427 us	67.94082 us	0	0.00059	0.00117
Q2	78.70943 us	137.75082 us	0	0.00037	0.00074
Q3	29.53585 us	46.85734 us	0	0.00043	0.00087
Q4	43.38019 us	17.80969 us	0	0.00029	0.00058

La siguiente imagen representa el mapa topológico del dispositivo, los ratios de error para las puertas CNOT, H, y el ratio de error de lectura.

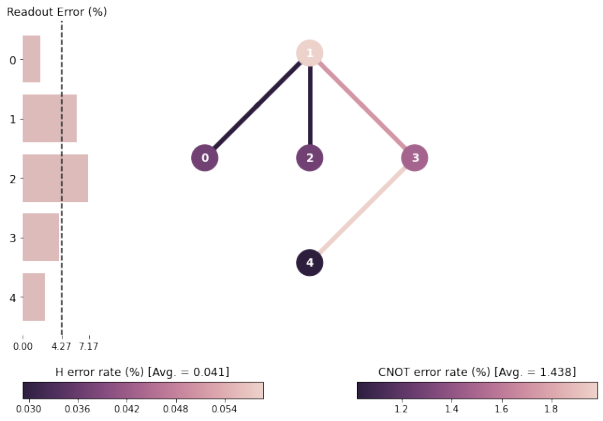


Figura 4.2: Mapa topológico del ordenador ibmq_london.

ibmq_16_melbourne

- **Número de qubits:** 15.
- **Volumen cuántico** 8.
- **Número máximo de ejecuciones** 8192.
- **Disponible desde:** 6-11-2018.

	T1	T2	Error puerta U1	Error puerta U2	Error puerta U2
Q0	64.43691 us	101.06709 us	0	0.00053	0.00107
Q1	56.57467 us	57.59258 us	0	0.00084	0.00169
Q2	64.153 us	95.26257 us	0	0.00078	0.00155
Q3	77.52336 us	16.72313 us	0	0.0005	0.001
Q4	53.12886 us	59.40107 us	0	0.00105	0.00209
Q5	20.99515 us	40.54361 us	0	0.0024	0.0048
Q6	62.13607 us	76.1998 us	0	0.0011	0.00221
Q7	33.71643 us	13.32344 us	0	0.00164	0.00328
Q8	100.20941 us	116.59563 us	0	0.00043	0.00085
Q9	36.30509 us	68.26763 us	0	0.00228	0.00455
Q10	85.86389 us	66.565 us	0	0.00135	0.00269
Q11	44.06823 us	79.13941 us	0	0.0005	0.00099
Q12	84.23371 us	75.58931 us	0	0.00068	0.00136
Q13	34.30024 us	32.00742 us	0	0.00202	0.00404
Q14	44.36103 us	56.64357 us	0	0.00064	0.00128

La siguiente imagen representa el mapa topológico del dispositivo, los ratios de error para las puertas CNOT, H, y el ratio de error de lectura.

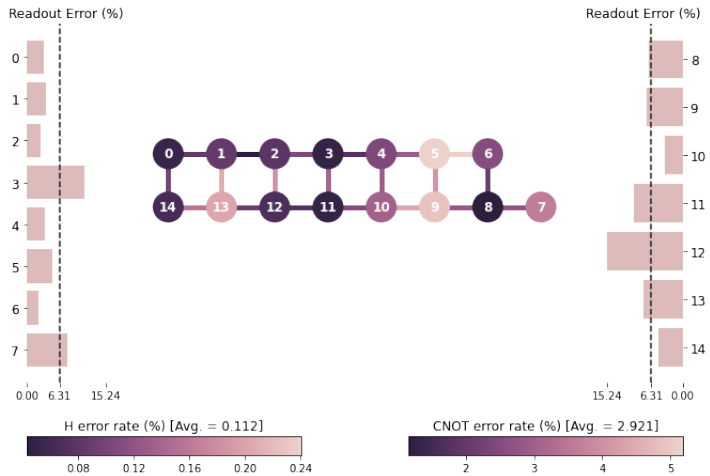


Figura 4.3: Mapa topológico del ordenador ibmq_16_melbourne.

4.2. Desarrollo de los programas con la herramienta Qiksit y ejecución en los computadores cuánticos de IBM

Qiskit es una herramienta creada por IBM para el desarrollado de Software cuántico. Su versión principal usa el lenguaje de programación Python y puede se importar en forma de librería. Permite crear y manipular programas cuánticos, además de ejecutarlos en dispositivos reales o simuladores. Usa el modelo de circuito para representar los programas que pueden ser ejecutados en cualquier dispositivo cuántico [1].

Primero importamos los paquetes necesarios.

```
from qiskit.compiler import transpile, assemble
from qiskit import QuantumCircuit, execute, Aer, IBMQ
from qiskit.tools.monitor import job_monitor
from qiskit.tools.jupyter import *
from qiskit.visualization import *
```

Podemos crear nuestros circuitos cuánticos con el tipo de dato **QuantumCircuit**.

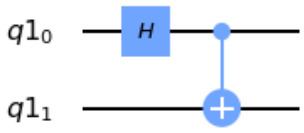
```
circuitoQ = QuantumCircuit()
```

Una vez tenemos nuestra estructura de circuito creada, pasamos a crear nuestras unidades cuánticas: los textbfqubits. Estos serán los registros de nuestro circuito.

```
Qbits = QuantumRegister(2)
circuitoQ.add_register(Qbits)
```

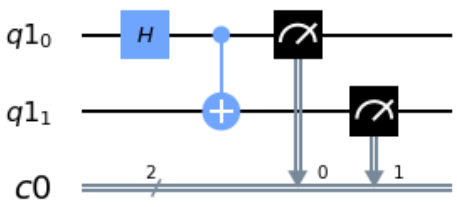
Podemos actuar sobre los qubits con puertas cuánticas. En este ejemplo vamos a probar con la Hardamard y Cnot. Para visualizar los circuitos usaremos el método **.draw()**

```
circuitoQ.h(0)
circuitoQ.cx(0,1)
circuitoQ.draw()
```



Podemos añadir **bits clásicos** los circuitos. Estos bits clásicos nos sirven para guardar el resultados de las observaciones. Para observar, es decir "medir", usaremos el método **measure(Qbit,bit)**.

```
circuitoQ.measure(0,0)
circuitoQ.measure(1,1)
circuitoQ.draw()
```

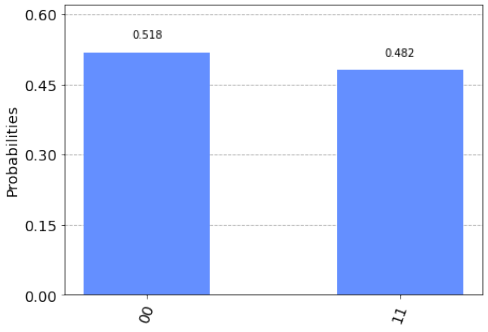


Ahora podemos simular este circuito de manera local con el simulador **qasm_simulator**. Los selecionaremos usando el comando **Aer.get_backend ('qasm_simulator')**. Se ejecuta con **execute(circuito,simulador,shots)**, y simulará el circuito tantas veces como especifiquemos en el parámetro **shots**.

```
emulador = Aer.get_backend('qasm_simulator')
job = execute(circuitoQ,emulador,shots=5000)
```

El resultado de la ejecución es un diccionario con el histograma de la simulación. Podemos visualizarlo con **plot_histogram(resultado)**.

```
resultado = job.result().get_counts()
plot_histogram(resultado)
```



Por último veremos como acceder a sistemas cuáticos reales para ejecutar nuestros circuitos. Para ello necesitamos una cuenta en IBM Quantum

experience. Esta cuenta es gratuita y cualquier persona puede obtener una. Debemos cargar nuestra cuenta, para ver los dispositivos a los que tenemos acceso.

```
provider = IBMQ.load_account()

for sistema in provider.backends():
    print(sistema)
```

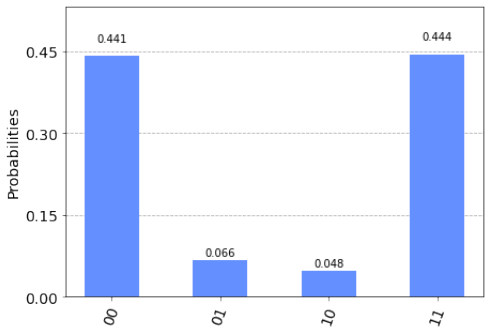
Salida: ibmqx2, ibmq_16_melbourne, ibmq_vigo, ibmq_ourense, ibmq_valencia, ibmq_london, ibmq_burlington, ibmq_essex, ibmq_armonk, ibmq_santiago.

Para usar los sistemas reales debemos crear un objeto con **provider.get_backend('dispositivo')**. Ejecutamos al igual que con el simulador, y podemos monitorizar la ejecución con **job_monitor(job)**.

```
sistema_real = provider.get_backend('ibmq_london')
job = execute(circuitoQ,backend=sistema_real, shots=5000)
job_monitor(job)
```

Podemos visualizar los resultados al igual que en el simulador.

```
resultado = job.result().get_counts()
plot_histogram(resultado)
```



Como podemos ver, hay un pequeño margen de error respecto a los resultados del simulador. Esto se debe a que el simulador simula las condiciones ideales del circuito que se corresponden al desarrollo matemático del mismo. Por el otro lado el dispositivo real no trabaja en condiciones ideales y los estados se ven perturbados debido a la decoherencia cuántica.

Capítulo 5

Resultados de la programación y ejecución de algoritmos en computadores cuánticos

En este capitulo, se implementarán en dispositivos cuánticos reales los algoritmos expuestos en el capítulo 3 y se comprobará la eficacia de los mismos. Para cada algoritmo, se presentará el código usado para crear los circuitos, y se utilizará el simulador 'qasm_simulator' para probar los algoritmos. Después se realizarán pruebas en los tres ordenadores cuánticos experimentales y se hará un análisis de los resultados. Según se ha indicado, el esquema que se seguirá es el siguiente:

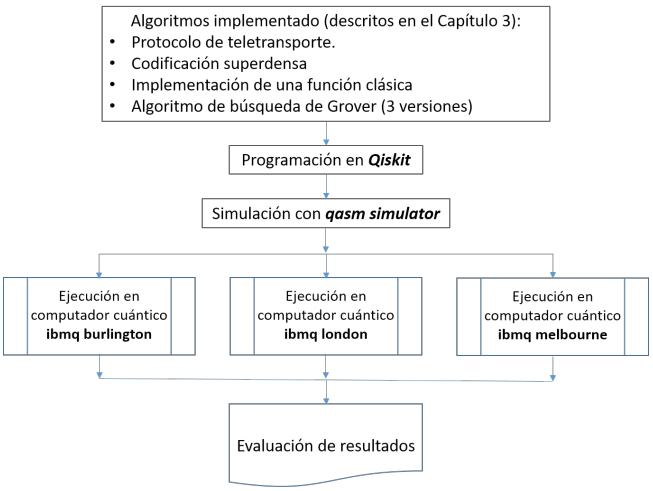


Figura 5.1: Esquema de experimentación.

5.1. Protocolo de teletransporte

A continuación, se presenta en lenguaje Qiskit el código usado para poder realizar el protocolo del teletransporte. El algoritmo utilizado se describe en la Secc. 3.3.2 de la presente memoria. El proceso se implementa con puertas Hadamard, Cnot, X y Z (ver Secc. 3.3.2)

Simulación del circuito

```
def crear_circuito_teletransporte(circuito,
Qubit_teletransportar, Qubit_receptor):

    #Definir registros necesarios
    Qubit_Enlace = QuantumRegister(1,'enlace')   #Qubit
    para entrelazar

    C1 = ClassicalRegister(1,'medida1')   #Registros Clá
    sicos
    C2 = ClassicalRegister(1,'medida2')
    C3 = ClassicalRegister(1,'medida3')

    #Añadir registros al circuito
    circuito.add_register(Qubit_teletransportar,
    Qubit_Enlace,Qubit_receptor,C1,C2,C3)

    #Crear entrelazamiento entre El Qubit receptor y el
    Qubit de enlace
    circuito.h(1)        #Puerta Hadamard sobre Qubit de
    enlace
    circuito.cx(1,2)     #Puerta Cnot sobre Qubit de
    enlace y Qubit receptor
    circuito.barrier()

    #Aplicar operadores de 'Alice'
    circuito.cx(0,1)     #Puerta Cnot sobre Qubit a
    teletrasnportar y Qubit de enlace
    circuito.h(0)        #Puerta Hadamard sobre Qubit a
    teletrasnportar
    circuito.barrier()

    #Medidas del sistema
    circuito.measure(0,0)    #medida del QUbit a
    teletransportar
    circuito.measure(1,1)    #medida del QUbit de enlace
    circuito.barrier()

    #Operadores de 'Bob'
```

```
29      circuito.z(2).c_if(C1,1)     #Puerta X sobre Qubit
        receptor, si medida teletransportar = 1
30      circuito.x(2).c_if(C2,1)     #Puerta Z sobre Qubit
        receptor, si medida teletransportar = 1
31
32      #medimos el resultado
33      circuito.measure(2,2)
34
35  ###
36  #creamos el circuito y los qubits necesarios
37  teletransportar = QuantumRegister(1,'teletransportar')
38  receptor = QuantumRegister(1,'receptor')
39  t = QuantumCircuit()
40
41  crear_circuito_teletransporte(t, teletransportar,
    receptor)
```

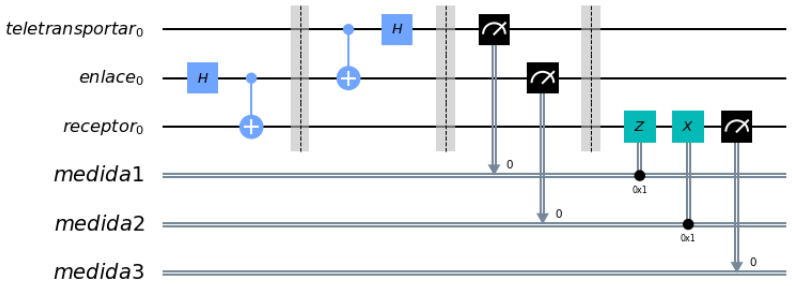


Figura 5.2: Visualización del circuito de teletransporte

Si probamos a ejecutar el circuito en el simulador 'qasm_simulator', obtenmos los siguienes resultados:

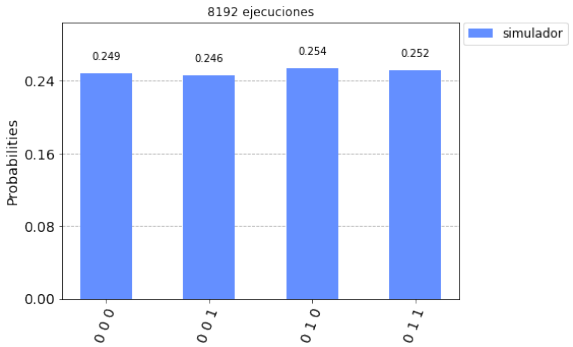


Figura 5.3: Resultados de la ejecución del circuito de la figura 3.1 en el simulador.En el eje 'x' se muestran los estados obtenidos.

Como podemos ver, para todos los estados obtenidos, el qubit receptor, se encuentra en el estado inicial del qubit emisor, en este caso $|0\rangle$.

Ejecución en computadores cuánticos reales

Debido a las limitaciones físicas de los dispositivos con los que trabajamos, no se pueden aplicar operadores en un circuito tras una medida. Por tanto, el circuito anteriormente presentado no se podría ejecutar. Para solucionar este problema, se puede crear un circuito que simule el comportamiento del circuito anterior usando los qubits que mide Alice como qubits de control para las siguientes operaciones. La puerta Z controlada se implementa internamente con dos puertas Hadamard y una CNOT. El circuito es el siguiente:

```
1  def crear_circuito_teletransporte(circuito,
   Qubit_teletransportar, Qubit_receptor):
2
3      #Definir registros necesarios
4      Qubit_Enlace = QuantumRegister(1,'enlace')   #Qubit
       para entrelazar
5
6      C1 = ClassicalRegister(1,'medida')   #Registro Clá
       sico
7
8
9      #Añadir registros al circuito
10     circuito.add_register(Qubit_teletransportar,
       Qubit_Enlace,Qubit_receptor,C1)
11
12     #Crear entrelazamiento entre El Qubit receptor y el
       Qubit de enlace
13     circuito.h(1)        #Puerta Hadamard sobre Qubit de
       enlace
14     circuito.cx(1,2)     #Puerta Cnot sobre Qubit de
       enlace y Qubit receptor
15     circuito.barrier()
16
17     #Aplicar operadores de 'Alice'
18     circuito.cx(0,1)     #Puerta Cnot sobre Qubit a
       teletrasnportar y Qubit de enlace
19     circuito.h(0)        #Puerta Hadamard sobre Qubit a
       teletrasnportar
20     circuito.barrier()
21
22     #Operadores de 'Bob'
23     circuito.cz(0,2)    #Puerta X sobre Qubit receptor,
       si medida teletransportar = 1
24     circuito.cx(1,2)    #Puerta Z sobre Qubit receptor,
       si medida teletransportar = 1
```

```
    circuito.barrier()

    #Medida del sistema
    circuito.measure(2,0)
#######
#creamos el circuito y los qubits necesarios
teletransportar = QuantumRegister(1,'teletransportar')
receptor = QuantumRegister(1,'receptor')
tr = QuantumCircuit()
crear_circuito_teletransporte(tr, teletransportar,
receptor)
```

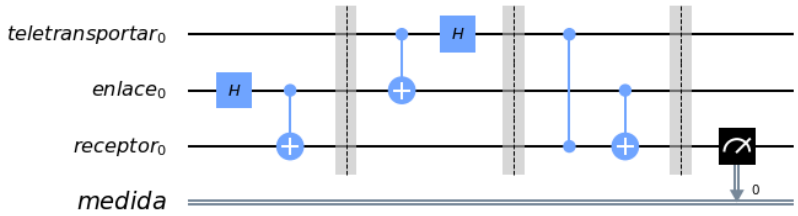


Figura 5.4: Visualización del circuito de para simular el teletransporte

N_Registros	N_Qubits	Puertas de un qubit	CNOT	Toffoli
4	3	4	4	0

Los resultados de ejecutar el circuito en los diferentes dispositivos son los siguientes:

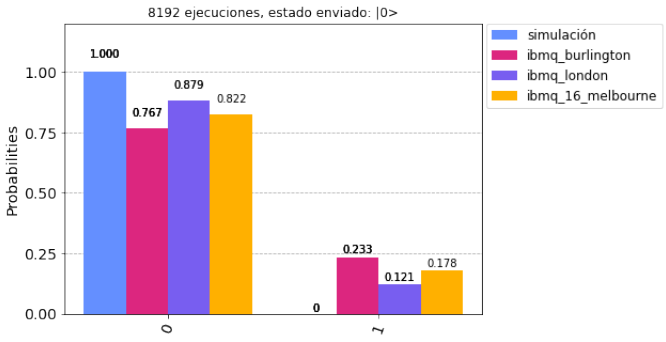


Figura 5.5: Resultados experimentales de la ejecución del circuito de la figura 3.3 para 8192 ejecuciones en cada dispositivo.

Los ratios de error experimental para cada dispositivo son los siguientes:

Dispositivo	ibmq_burlington	ibmq_london	ibmq_16_melbourne
ratio de error	23.254 %	12.060 %	17.785 %

5.2. Codificación superdensa

A continuación, se presenta en lenguaje Qiskit el código para realizar codificación superdensa. El algoritmo utilizado se describe en la Secc. 3.3.3 de la presente memoria. El proceso se implementa con puertas Hadamard, Cnot, X y Z (ver Secc. 3.3.3)

```
def crear_circuito_codificacion(circuito, mensaje):

    #Definir registros necesarios
    Qubit_codificador = QuantumRegister(1,'alice')   #
    Qubit para codificar
    Qubit_decodificar = QuantumRegister(1,'bob')

    C1 = ClassicalRegister(1,'medida1')   #Registros Clá
    sicos
    C2 = ClassicalRegister(1,'medida2')

    #Añadir registros al circuito
    circuito.add_register(Qubit_codificador,
    Qubit_decodificar,C1,C2)

    #Crear entrelazamiento
    circuito.h(0)        #Puerta Hadamard
    circuito.cx(0,1)     #Puerta Cnot sobre
    circuito.barrier()

    #Aplicar operadores de 'Alice'
    if(mensaje == "00"):
        pass
    elif (mensaje == "01"):
        circuito.z(0)
    elif (mensaje == "10"):
        circuito.x(0)
    elif (mensaje == "11"):
        circuito.z(0)
        circuito.x(0)
    circuito.barrier()

    #Decodificación
    circuito.cx(0,1)     #Puerta Cnot
    circuito.h(0)        #Puerta Hadamard

    #Medidas del sistema
    circuito.measure(0,0)
    circuito.measure(1,1)
######
#Creamos un objeto circuito y definimos el mensaje
```

```
39 coding = QuantumCircuit()
40 mensaje= "11"
41 crear_circuito_codificacion(coding, mensaje)
```

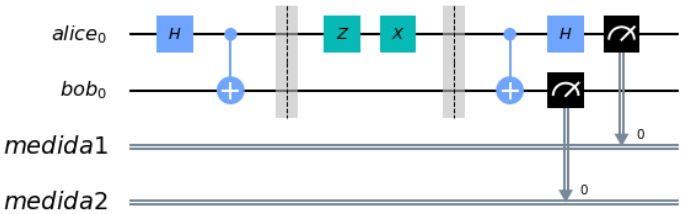


Figura 5.6: Visualización del circuito de codificación superdensa para el mensaje '11'.

El circuito resultante es el que probaremos tanto en el simulador como en los dispositivos cuánticos.

Simulación del circuito

Para este circuito el simulador reproduce las condiciones ideales de la evolución del sistema, y obtenemos el resultado previsto.

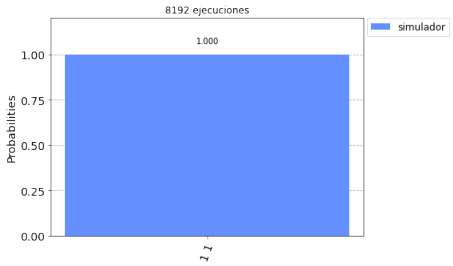


Figura 5.7: Resultados de la ejecución del circuito de la figura 3.5 en el simulador.

Ejecución en computadores cuánticos reales

N_Registros	N_Qubits	Puertas de un qubit	CNOT	Toffoli
4	2	4	2	0

Los resultados de ejecutar el circuito en los diferentes dispositivos son los siguientes:

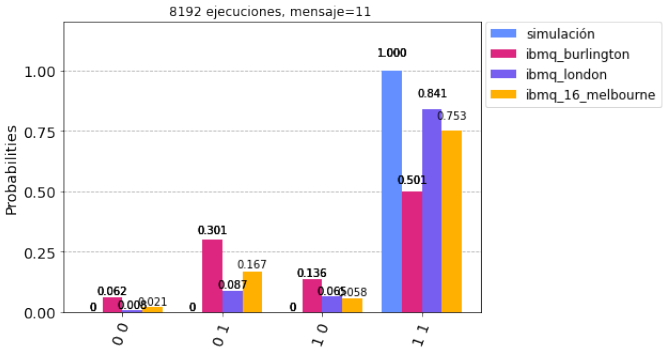


Figura 5.8: Resultados experimentales de la ejecución del circuito de la figura 3.5 para 8192 ejecuciones en cada dispositivo.

Los ratios de error experimental para cada dispositivo son los siguientes:

Dispositivo	ibmq_burlington	ibmq_london	ibmq_16_melbourne
ratio de error	49.853 %	15.917 %	24.743 %

5.3. Implementación de una función clásica: sumador completo de 2 bits

A continuación, se presenta en lenguaje Qiskit el código usado para poder crear un circuito que implementa un sumador completo de 2 bits. El proceso se implementa con puertas tofoli y cnot (la puerta cnot, puede ser simulada con toffoli, por cuestiones de eficiencia se usa directamente la cnot)(ver Secc. 3.4.2). En este caso 'A' y 'B' corresponde con los bits a sumar, 'C' es el acareo en la entrada y el resultado en la salida, y 'S' el acarreo en la salida.

```
#definición de registros
def crear_resgistros_sumador(circuito):
    #definimos los Qubits de nuestro sistema
    A = QuantumRegister(1,'a')
    B = QuantumRegister(1,'b')
    C = QuantumRegister(1,'c')
    S = QuantumRegister(1,'s')
    #definimos los registros clásicos
    M = ClassicalRegister(4,'medida')
    circuito.add_register(A,B,C,S,M)
#inicializa los bits de entrada
def inicializar_clasico(circuito,C,B,A):
    if((A!=1 and A!=0) or (B!=1 and B!=0) or (C!=1 and C
    !=0)):
        print("valores erroneos")
    else:
        if(A==1):
            circuito.x(0)
        if(B==1):
            circuito.x(1)
        if(C==1):
            circuito.x(2)
#crea el circuito que implementa la función de suma
def crear_circuito_sumador(circuito):
    circuito.barrier()
    #Puertas cnot y tofoli aplicadas al circuito
    circuito.ccx(0,1,3)
    circuito.cx(0,1)
    circuito.ccx(1,2,3)
    circuito.cx(1,2)
    circuito.cx(0,1)
    circuito.barrier()
n=4 #número de qubits
sumador = QuantumCircuit()
crear_resgistros_sumador(sumador)
inicializar_clasico(sumador,0,0,1) #entrada A=1 B=0 C=0
crear_circuito_sumador(sumador)
sumador.measure(range(n),range(n))
```

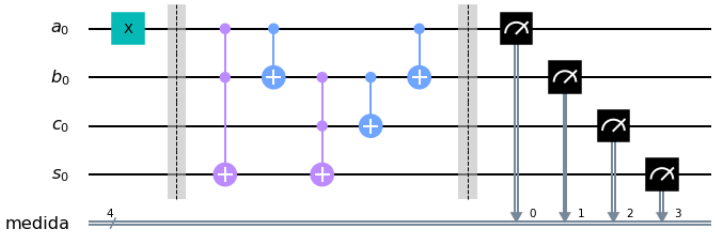


Figura 5.9: Visualización del circuito del sumador para la entrada A=1 B=0 C=0 S=0.

Simulación del circuito

Si realizamos una simulación del circuito anterior podremos ver que el resultado es A=1 B=0 C=1 S=0.

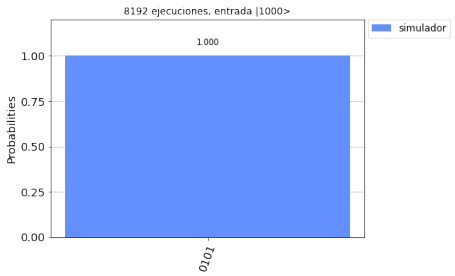


Figura 5.10: Resultados de la simulación del circuito de la figura 3.8.

Para ver el potencial que tienen los sistemas cuánticos, podemos poner los dos primeros bits de entrada en una superposición equiprobable y simular el circuito.

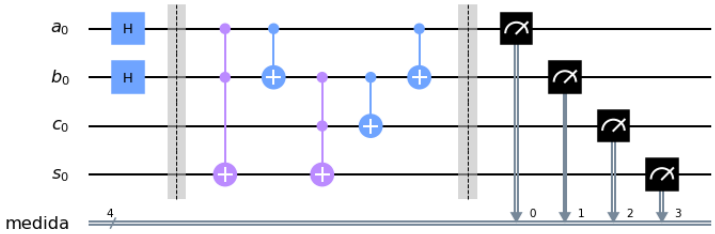


Figura 5.11: Visualización del circuito del sumador para la entrada A=$\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ B=$\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ C=0 S=0.

La siguiente gráfica, obtenida a partir de ejecuciones en el simulador, representa cuales serían las amplitudes del sistema antes de realizar la medida. Como se puede observar, se computan todas las posibles entradas(de los qubits en superposición) en paralelo.

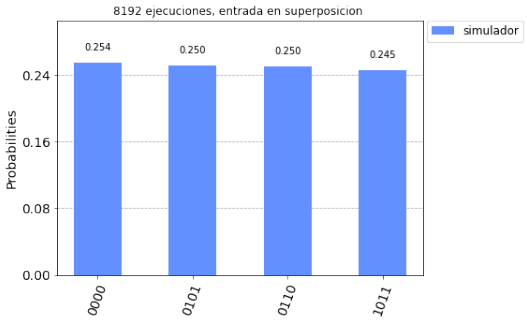


Figura 5.12: Resultados de la simulación del circuito de la figura 3.10.

Ejecución en dispositivos reales

Para ejecutar el circuito en los dispositivos reales, modificaremos el circuito para obtener como salida el resultado del computo en un qubit auxiliar y el estado inicial del sistema (tal y como se explica en la sección 2.4 de circuitos clásicos).

$$U_C \left|x\right\rangle \left|0\right\rangle = \left|x\right\rangle \left|C(x) \oplus 0\right\rangle .$$

Para ello aplicaremos el circuito inverso al sumador y añadiremos un qubit extra.

```
def crear_circuito_sumador_invertido(circuito):
    #Definimos los registros auxiliares
    Y = QuantumRegister(1,'y')
    F = ClassicalRegister(1,'f(x)')
    circuito.add_register(Y,F)

    #enviamos la información al qubit auxiliar
    circuito.cx(2,4)
    circuito.barrier()
    #aplicamos el circuito inverso
    circuito.cx(0,1)
    circuito.cx(1,2)
    circuito.ccx(1,2,3)
    circuito.cx(0,1)
    circuito.ccx(0,1,3)
    circuito.barrier()
```

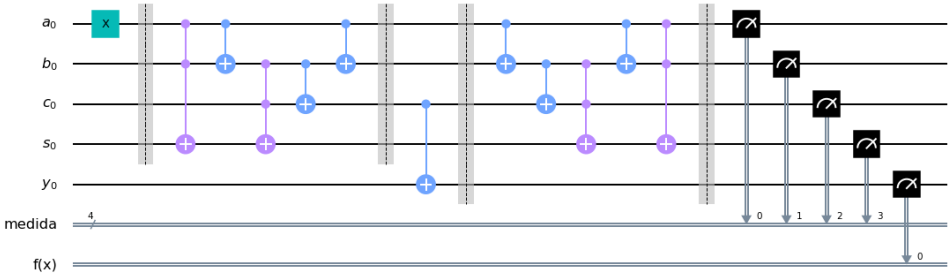


Figura 5.13: Visualización del circuito del sumador modificado.

N_Registros	N_Qubits	Puertas de un qubit	CNOT	Toffoli
10	5	1	7	4

Los resultados de ejecutar el circuito en los diferentes dispositivos son los siguientes:

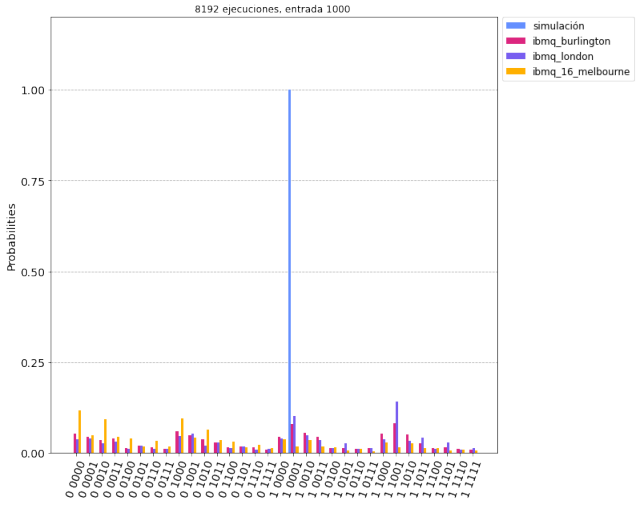


Figura 5.14: Resultados experimentales de la ejecución del circuito de la figura 3.12 para 8192 ejecuciones en cada dispositivo

Los ratios de error experimental para cada dispositivo son los siguientes:

Dispositivo	ibmq_burlington	ibmq_london	ibmq_16_melbourne
ratio de error	92.028 %	89.880 %	98.181 %

Para este tipo de circuitos que requieren de más registros y operaciones, hay demasiado ruido. Esto se debe a que se sobrepasan los tiempos de decoherencia durante la ejecución. Además, el número de operaciones físicas para implementar puertas complejas como la Toffoli, se dispara, y se van acumulando muchos errores por operación.

5.4. Algoritmo de Grover

En esta sección se presentan diferentes versiones del algoritmo de Grover, se describe en la Secc. 3.5 de la presente memoria.. Para evitar exceso errores en los resultados experimentales al ejecutar circuitos muy extensos, se presentarán los circuitos con una iteración del protocolo de amplificación.

5.4.1. 6 qubits, 3 qubits de entrada, función Clásica

Para empezar se presenta una versión para un espacio de busqueda de $2^3 = 8$ estados implementada con 3 qubits para codificar la entrada y otros 3 que sirven para implementar la función de busqueda y el oráculo.

Para el oráculo se implementa una función 'AND' de 3 entradas usando puertas Toffoli, el estado solución será $|111\rangle$. El oraculo se implementa como es descrito anteriormente en el apartado del algoritmo de búsqueda

$$U_f |x\rangle |-\rangle = |x\rangle |f(x) \oplus |-\rangle\rangle .$$

Para el operador de difusión, aplicaremos puertas Hadamard sobre los qubits de entrada, implementaremos un circuito que realice una reflexión del vector de estado sobre el estado $|000\rangle$, y volveremos a aplicar puertas Hadamard. Es decir, implementamos la siguiente operación:

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

Para implementar la reflexión sobre $|000\rangle$, implementaremos una puerta Z doblemente controlada, solo para el estado $|111\rangle$ realiza un cambio de signo en la amplitud del sistema. Como queremos que el estado sea el $|000\rangle$, rodearemos el circuito con Puertas X.

El siguiente código es usado para generar el circuito:

```
n=3# numero de qubits en la entrada de la función
#Creamos el circuito Oraculo
def oraculo(circuito):
    circuito.barrier()
    #inicializamos el qbit auxiliar
    circuito.x(5)
    circuito.h(5)
    #funcion and de 3 bits
    circuito.ccx(0,1,3)
    circuito.ccx(2,3,4)
    #pasamos el resultado
    circuito.cx(4,5)
    #inversa de funcion and de 3 bits
    circuito.ccx(2,3,4)
    circuito.ccx(0,1,3)
#Creamos el circuito del operador de difusion
def difusion(circuito):
```

```
18      #Aplicamos hadamard y X a cada qubit
19      for q in range(n):
20          circuito.h(q)
21          circuito.x(q)
22      #simulamos una puerta Z doblemente controlada
23      circuito.barrier()
24      circuito.h(2)
25      circuito.ccx(0,1,2)
26      circuito.h(2)
27      circuito.barrier()
28      #Aplicamos X y hadamard a cada qubit
29      for q in range(n):
30          circuito.x(q)
31          circuito.h(q)
32
33  grover3 = QuantumCircuit()
34  A = QuantumRegister(6)
35  M = ClassicalRegister(3,'medida')
36  grover3.add_register(A,M)
37  #inicializamos a un estado de superposición equiprobable
38  grover3.h(range(n))
39  #aplicamos el oráculo y el operador de difusión
40  oraculo(grover3)
41  difusion(grover3)
42  #medimos
43  grover3.measure(range(n),range(n))
```

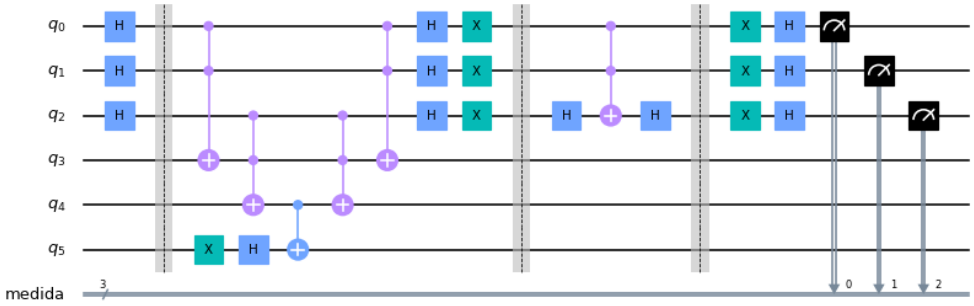


Figura 5.15: Visualización del circuito de Grover con 6 qubits para una iteración del protocolo de amplificación.

Como podemos observar el algoritmo consta de 3 pasos:

1. Inicialización de la entrada a una superposición equiprobable.
2. Aplicación del circuito oráculo.
3. Aplicación del circuito de difusión.

Simulación del circuito

Simulando el circuito con una sola iteración obtenemos los siguientes resultados:

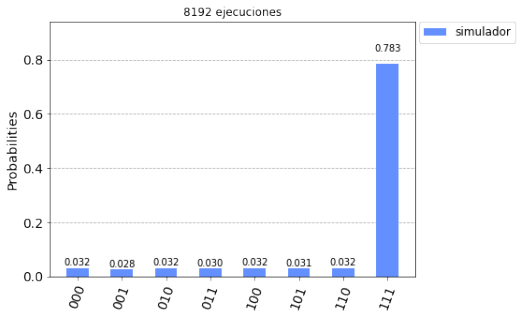


Figura 5.16: Resultados de la ejecución del circuito de la figura 3.14 en el simulador.

El circuito amplifica la amplitud del estado $|111\rangle$, valores solución de nuestra función 'AND'.

Ejecución en dispositivos reales

N_Registros	N_Qubits	Puertas de un qubit	CNOT	Toffoli
9	6	19	1	5

El único dispositivo con más de 5 qubits es 'ibmq_16_melbourne', los resultados en este dispositivo son los siguientes:

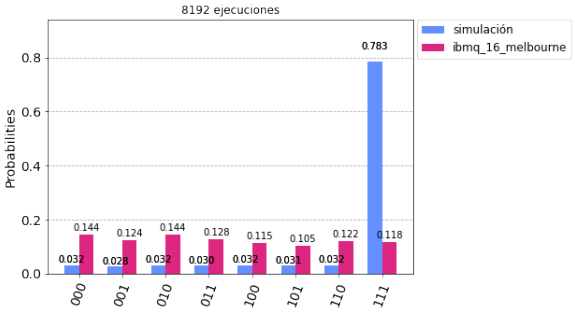


Figura 5.17: Resultados de la ejecución del circuito de la figura 3.14 en el dispositivo ibm_16_melbourne.

El ratio de error es el siguiente:

Dispositivo	ibmq_16_melbourne
ratio de error	84.99 %

Para este circuito con tantos qubits y puertas el ruido es demasiado alto.

5.4.2. 3 qubits Optimización

Los circuitos demasiado complejos no aportan buenos resultados en los dispositivos reales. Para reducir el la complejidad el circuito anterior, podemos condensar el oráculo en un circuito para 3 qubits que haga la misma función. En este caso no estaremos implementando la función que Clásica que queramos, pero podemos elegir que estado amplificar. Para ello usaremos la puerta Z doblemente controlada que se usa en la difusión, este circuito es mucho más sencillo y también invierte la amplitud del sistema para el estado $|111\rangle$.

```
1 def oraculo(circuito):
2     circuito.h(2)
3     circuito.ccx(0,1,2)
4     circuito.h(2)
```

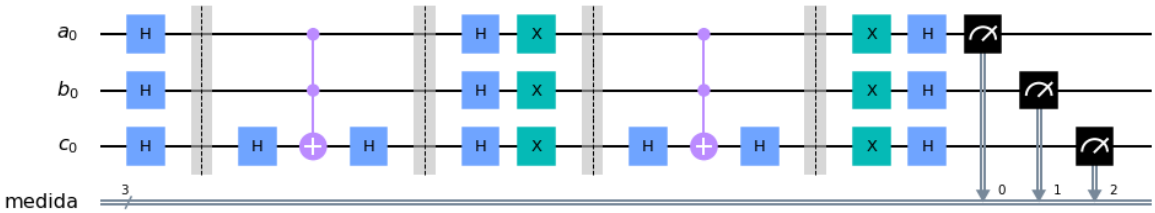


Figura 5.18: Visualización del circuito de Grover con 3 qubits para una iteración del protocolo de amplificación.

Simulación del circuito

Simulando el circuito obtenemos los siguientes resultados:

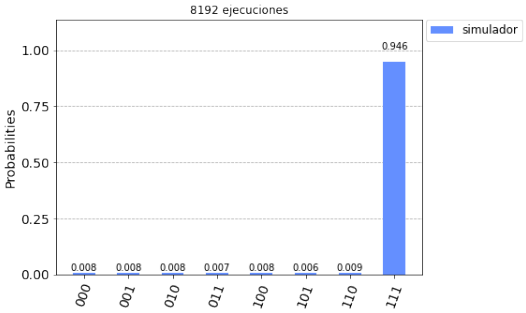


Figura 5.19: Resultados de la ejecución del circuito de la figura 3.17 en el simulador.

Se obtiene un pequeño margen de error porque estamos probando el algoritmo para entradas muy pequeñas.

N_Registros	N_Qubits	Puertas de un qubit	CNOT	Toffoli
6	3	19	0	2

Los resultados de ejecutar el circuito en los diferentes dispositivos son los siguientes:

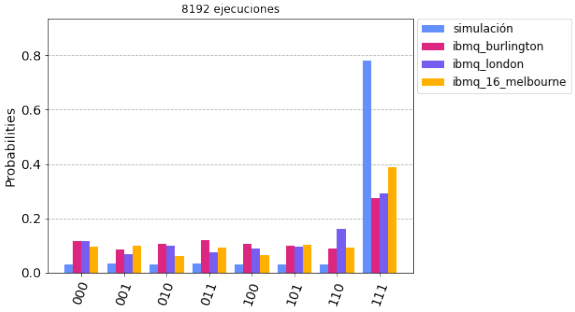


Figura 5.20: Resultados experimentales de la ejecución del circuito de la figura 3.17 para 8192 ejecuciones en cada dispositivo

Los ratios de error experimental para cada dispositivo son los siguientes:

Dispositivo	ibmq_burlington	ibmq_london	ibmq_16_melbourne
ratio de error	64.675 %	62.37 %	50.329 %

Como podemos ver, al reducir el número de qubits, ganamos algo de precisión.

5.4.3. 5 qubits, 4 qubits de entrada

Esta versión, simplemente escala la versión anterior, con la particularidad de que para implementar una puerta not triplemente controlada(CCCNOT), necesitamos un qubit auxiliar [14]. Para implementar la CCCNOT se usan 3 puertas Toffoli. La puerta CCCNOT se utiliza como en el circuito anterior para simular una puerta Z triplemente controlada. Para este circuito el estado solución es $|0111\rangle$, por tanto añadiremos una puerta X al principio y al final del circuito. Aplicando puertas X en cada qubit, podemos controlar el estado que queremos obtener.

```
n=4 #numero de qubits
grover4 = QuantumCircuit()
Q = QuantumRegister(5,'q')
M = ClassicalRegister(5,'medida')
grover4.add_register(Q,M)

grover4.h(range(n)) #entrada en superposición
grover4.barrier()
#Aplicamos el oráculo
grover4.x(3)
grover4.ccx(0,1,4) #simulación de cccx
grover4.ccx(4,2,3)
grover4.ccx(0,1,4)

grover4.h(3)

grover4.ccx(0,1,4) #simulación de cccx
grover4.ccx(4,2,3)
grover4.ccx(0,1,4)
grover4.x(3)
###Aplicamos el operador de difusión
grover4.barrier()

grover4.h(range(n))
grover4.x(range(n))
grover4.ccx(0,1,4) #simulación de cccx
grover4.ccx(4,2,3)
grover4.ccx(0,1,4)

grover4.h(3)

grover4.ccx(0,1,4) #simulación de cccx
grover4.ccx(4,2,3)
grover4.ccx(0,1,4)
grover4.x(range(n))
grover4.h(range(n))
```

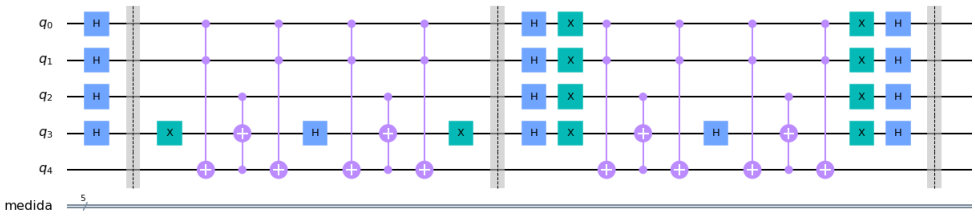


Figura 5.21: Visualización del circuito de Grover con 5 qubits para una iteración del protocolo de amplificación.

Simulación del circuito

Simulando el circuito obtenemos los siguientes resultados:

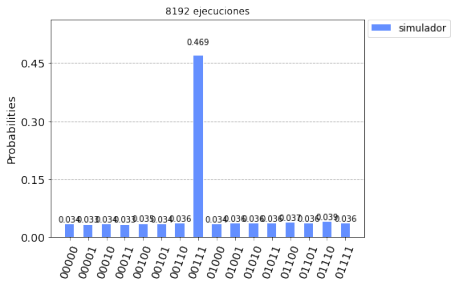


Figura 5.22: Resultados de la ejecución del circuito de la figura 3.20 en el simulador.

Ejecución en dispositivos reales

N_Registros	N_Qubits	Puertas de un qubit	CNOT	Toffoli
9	5	24	0	12

Los resultados de ejecutar el circuito en los diferentes dispositivos son los siguientes:

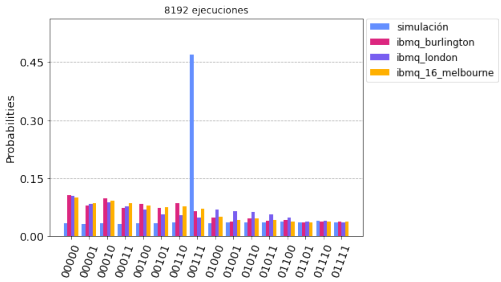


Figura 5.23: Resultados experimentales de la ejecución del circuito de la figura 3.17 para 8192 ejecuciones en cada dispositivo

Los ratios de error experimental para cada dispositivo son los siguientes:

Dispositivo	ibmq_burlington	ibmq_london	ibmq_16_melbourne
ratio de error	85.948 %	89.487 %	84.829 %

5.5. Evaluación de los resultados

Se puede observar que obtenemos resultados dispares para un mismo circuito, esto se debe a que la arquitectura de cada ordenador cuántico juega un papel fundamental en la ejecución. Los resultados varían dependiendo de los ratios de error que experimenta cada dispositivo en las operaciones físicas, de su topología y de los tiempos de decoherencia.

Para circuitos pequeños se obtienen resultados razonables. En estos casos, el dispositivo que mejores resultados ofrece es 'ibmq_london', que es el que tiene mayor volumen cuántico. El siguiente dispositivo con mejores prestaciones es 'ibm_16_melbourne', aunque tenga el mismo volumen que 'ibmq_burlington', al contar con más qubits, exiten más posibilidades de implementar cada circuito a nivel de hardware.

Para circuito más extensos, los resultados acumulan demasiados errores y no son aceptables, obtenemos ratios de erorr por encima del 50 %. Cabe destacar que el dispositivo 'ibm_16_melbourne', ofrece resultados ligeramente mejores por contar con más qubits, al igual que en el caso anterior.

Con estos resultados, los ordenadores cuánticos presentados, no se pueden contemplar para un uso convencional. Sin embargo, actualmente son capaces de ejecutar satisfactoriamente pequeños circuitos que son suficiente tanto para propósitos educativos, como para fines de investigación en todas las áreas de conocimiento circundantes a la computación cuántica y sus aplicaciones.

Parte IV

CONCLUSIONES Y VÍAS FUTURAS

Capítulo 6

Consecución de objetivos, conclusiones y vías futuras

Consecución de objetivos

En este trabajo se han explorado las posibilidades reales y prácticas, en el momento actual, de la computación cuántica en aplicaciones que se adjetivan como asombrosas y que son de incidencia en las telecomunicaciones cual es el caso del teletransporte como método de transmisión de información cuántica, la implementación de protocolos de transmisión de información clásica usando canales seguros de información cuánticos, o la implementación del algoritmo de búsqueda de Grover como mejora de los algoritmos clásicos de optimización.

Para poder entender los conceptos utilizados en el desarrollo de la Memoria, sin tener necesidad de un conocimiento previo sobre el tema o de acudir a fuentes que se encuentran dispersas en la bibliografía, dentro del Apartado II. Marco Teórico se ha incluido una recopilación original que muestra una introducción a los principios de la computación cuántica imprescindibles para entender el resto de la presente Memoria. También, dentro de dicho apartado, se ha realizado una descripción clara y rigurosa, a la vez que concisa, de la aplicación de los sistemas cuánticos a la computación, mostrando el origen de la gran potencia de cálculo obtenible con ellos.

En efecto, se han presentado de forma teórica protocolos que transmiten información usando las correlaciones cuánticas entre sistemas entrelazados. El protocolo de teletransporte es capaz de transmitir información cuántica entre sistemas que no están en contacto entre si. Este protocolo sirve como alternativa a la copia de estados cuánticos, que como se ha explicado, según el teorema de la no clonación, es imposible. Además es base para el desarrollo de las futuras tecnologías de comunicación cuántica como el internet cuántico [10]. El protocolo de codificación superdensa, demuestra la capacidad que tienen los sistemas cuánticos de comprimir y transmitir información clásica usando estados entrelazados de Bell. Este protocolo pone de manifiesto las

posibilidades de los canales de comunicación cuánticos para transmitir información clásica de manera segura, además este puede ser escalado usando sistemas entrelazados de mayor dimensión [9].

También se ha expuesto la capacidad de que tienen los ordenadores cuánticos para realizar cualquier función implementable en un ordenador clásico. Esto es fundamental para codificar problemas clásicos como puede ser una búsqueda. Se ha explicado el funcionamiento del algoritmo de Grover, que permite realizar una búsqueda en un conjunto desordenado de datos en un tiempo $O(\sqrt{2^n})$. Esto supone una mejora cuadrática respecto a la versión clásica del algoritmo de búsqueda para una secuencia de datos no ordenada. Este algoritmo juega un papel fundamental para la resolución de problemas NP-completos que son de gran relevancia en la mayoría de campos de la informática y las ingenierías.

En la parte experimental del trabajo, se han conseguido implementar con éxito los apartados expuestos anteriormente usando la herramienta IBM Quantum Experience. Tras la implementación se ha realizado una simulación satisfactoria de cada circuito para verificar su funcionamiento. Posteriormente se han realizado numerosas ejecuciones de cada algoritmo en tres ordenadores cuánticos diferentes que proporciona la herramienta, el nombre de estos dispositivos es Burlington, London y Melbourne, y sus características técnicas se presentan en el apartado 4.1. Tras realizar varias ejecuciones en cada dispositivo, se han calculado los ratios de error de cada uno en ellos.

Se ha ejecutado una versión adaptada al hardware utilizado del protocolo de teletransporte, y se ha conseguido transmitir el estado cuántico de un qubit con tan solo un ratio de error del 12.06 % en el mejor de los casos usando el ordenador cuántico de London.

Se ha ejecutado una versión del protocolo de codificación superdensa para codificar el mensaje binario '11', y se ha conseguido transmitir el mensaje con un ratio de error del 15.9 % en el mejor de los casos usando el ordenador cuántico de London.

Se ha implementado un sumador completo de dos bits usando circuitos cuánticos con 5 qubits, y no se han conseguido resultados aceptables al ejecutarlo en los ordenadores cuánticos reales, obteniendo un ratio de error medio superior al 90 %.

Se han implementado tres versiones diferentes del algoritmo de Grover. La primera usa 6 qubits para realizar una búsqueda sobre 8 estados posibles, tras las ejecuciones en el ordenador cuántico Melbourne (que es el único con más de 5 qubits disponibles) obtenemos un ratio de error de 84.99 %. La segunda versión pretende mejorar la primera, y para el mismo espacio de búsqueda, reduce el número de qubits usados en 3, tras las ejecuciones se obtiene un ratio de error del 50.33 % en el mejor de los casos usando el ordenador cuántico Melbourne. La última versión usa 5 qubits para realizar una búsqueda sobre 16 estados, tras las ejecuciones obtenemos un ratio de error medio por encima del 85 %.

Tras evaluar todas las ejecuciones, se ha conseguido observar que los dispositivos proporcionados por la herramienta cumplen su función con margenes de error aceptables para circuito sencillos y con pocos qubits, como el teletransporte, la codificación superdensa o incluso optimizaciones del algoritmo de Grover para espacios de búsqueda pequeños. Para circuitos más complejos y con mayor número de qubits, los ratios de error son demasiado altos, esto se debe a que se acumulan errores al aplicar muchos operadores (puertas lógicas cuánticas) sobre un circuito, y a que los tiempos de decoherencia de estos dispositivos son mayores o se aproximan a los tiempos de ejecución.

Conclusiones

Se ha comprobado que los dispositivos experimentales probados, son de gran utilidad para propósitos educativos o de investigación, pero no proporcionan resultados razonables para resolver problemas reales, ni mejoran las prestaciones de los computadores clásicos actuales. Podemos decir que el estado actual de los esfuerzos para construir una computadora cuántica universal, programable y funcional está en sus primeras etapas; como lo estaba la computación clásica en la primera mitad del siglo XX. Aunque teóricamente esté probado el gran potencial de los computadores cuánticos para resolver problemas complejos, transmitir información y simular sistemas cuánticos, el avance en este campo está limitado por el desarrollo de los dispositivos a nivel físico. Uno de los mayores problemas es el aislamiento. Cuanto mayor sea el procesador, es más difícil de aislar. El acceso a la potencia de cómputo que ofrecen los computadores cuánticos tiene el precio de la fragilidad. Son sistemas muy inestables, que tienden a perder rápidamente su coherencia cuántica y volver a las descripciones clásicas, debido principalmente a la interacción con el entorno, que tiende a destruir la superposición y el entrelazamiento [18].

En enero de 2019, la revista *Proceedings of the IEEE*, incluyó en su artículo "An outlook for quantum computing"[13] la siguiente figura (ver figura 6.1):

Esta figura proporciona un mapa sobre el estado de los computadores cuánticos y su capacidad computacional. En la figura se representa número de qubits frente a probabilidad de error de cada puerta.

La zona verde corresponde a la situación de septiembre de 2018: se dispone de sistemas de varios o a lo sumo de decenas de qubits y la probabilidades de error por puerta son del orden de 10^{-3} o superiores. Por debajo de la zona azul se encuentra la región donde la computación clásica es superior a la cuántica. La zona purpura se corresponde con la región de supremacía cuántica, esta región corresponde a donde la computación cuántica supera en potencia de cálculo a los grandes supercomputadores actuales. Según los investigadores de este artículo y como se muestra en la figura, se requieren

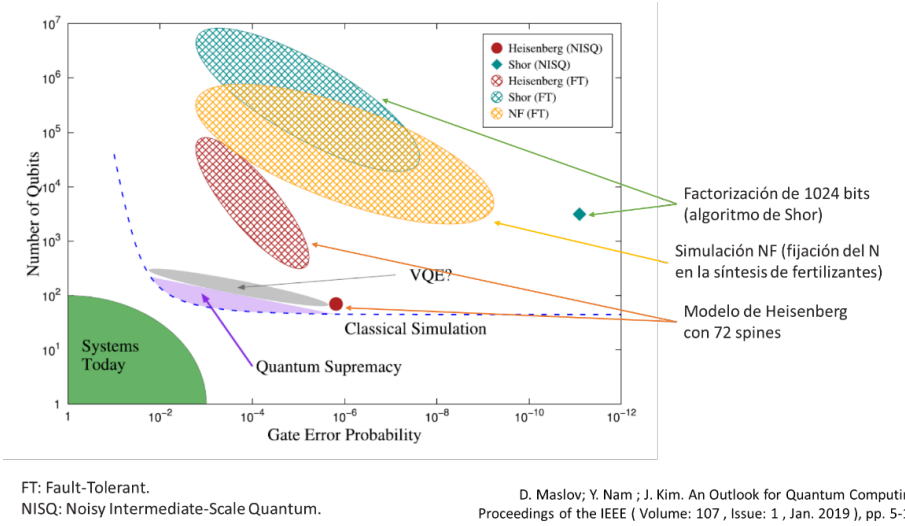


Figura 6.1: Estado del arte y potencial de los computadores cuánticos (2018).

arquitecturas del orden de 100 qubits con una probabilidad de error de funcionamiento por puerta menor de una centésima. Más tarde ese mismo año, investigadores de Google afirmaron haber alcanzado la zona de supremacía con una computadora cuántica que trabaja con 53 qubits [2].

Las zonas con entramado en color corresponden a las regiones donde la aplicación de distintos algoritmos tendrán interés práctico y podrán ser aplicados a problemas del mundo real. El diamante cián y la región sombreada correspondientes a la factorización de 1024 bits utilizando el algoritmo de Shor, el círculo magenta y la región sombreada representan la simulación de un modelo de Heisenberg de 72 espines, y la región sombreada en naranja ilustra la simulación NF (fijación del nitrógeno en la síntesis de fertilizantes).

A pesar de todo este potencial, por lo general, no se vislumbra el uso de la computación cuántica para aplicaciones de uso cotidiano (emails, generar documentos, etc.) sino como complemento y apoyo a supercomputadores [17].

Vías futuras

Las principales lineas de trabajo futuras para este proyecto son:

- Investigación para optimizar los presenten algoritmos y reducir su complejidad a nivel de circuito.
- Estudio a fondo de cada ordenador cuántico probado para poder realizar implementaciones a bajo nivel, atendiendo a las características técnicas de cada dispositivo.

- Investigación sobre protocolos de corrección de errores cuánticos para mitigar el efecto de la decoherencia cuántica en las ejecuciones.
- Estudio de diferentes algoritmos cuánticos que gran impacto en el ámbito de las ingenierías y las telecomunicaciones como el algoritmo de Shor, el algoritmo de temple cuántico, o el algoritmo cuántico para resolver sistemas de ecuaciones lineales.
- Estar atentos en todo momento al desarrollo de esta nueva tecnología ya que, como ha ocurrido con otras muchas, acabará obteniendo su madurez en cuyo momento será una herramienta transcendental de aplicación en el desarrollo de la ciencia.

Bibliografía

[1] Abraham Asfaw, Luciano Bello, Yael Ben-Haim, Sergey Bravyi, Lauren Capelluto, Almudena Carrera Vazquez, Jack Ceroni, Richard Chen, Albert Frisch, Jay Gambetta, Shelly Garion, Leron Gil, Salvador De La Puente Gonzalez, Francis Harkins, Takashi Imamichi, David McKay, Antonio Mezzacapo, Zlatko Minev, Ramis Movassagh, Giacomo Nannicni, Paul Nation, Anna Phan, Marco Pistoia, Arthur Rattew, Joachim Schaefer, Javad Shabani, John Smolin, Kristan Temme, Madeleine Tod, Stephen Wood, James Wootton. (2020) Learn Quantum Computation Using Qiskit. Recuperado de http://community.qiskit.org/textbook .

[2] Arute, F., Arya, K., Babbush, R., Bacon, D., Bardin, J. C., Barends, R., ... & Burkett, B. (2019). Quantum supremacy using a programmable superconducting processor. Nature, 574(7779), 505-510.

[3] Benenti, G., Casati, G., & Strini, G. (2004). Principles of Quantum Computation and Information-Volume I: Basic Concepts. World scientific.

[4] Díaz Rodríguez, Á., Prieto Espinosa, A. (2003). Arquitecturas de computación basadas en dispositivos de puntos cuánticos. Enlace: https://digibug.ugr.es/handle/10481/59973

[5] Ellwanger, D. (2019). A Quantum Computing Pamphlet.

[6] Einstein, A. (1948). Quanten-mechanik und wirklichkeit. Dialectica, 2(3-4), 320-324.

[7] Feynman, R. P. (1965). The Feynman Lectures on Physics Vol III. Narosa.

[8] Horsman, C., Stepney, S., Wagner, R. C., & Kendon, V. (2014). When does a physical system compute?. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences, 470(2169), 20140182.

[9] Hu, X. M., Guo, Y., Liu, B. H., Huang, Y. F., Li, C. F., & Guo, G. C. (2018). Beating the channel capacity limit for superdense coding with entangled ququarts. Science advances, 4(7), EAAT9304.

[10] Kimble, H. J. (2008). The quantum internet. Nature, 453(7198), 1023-1030.

[11] Lloyd, S. (2006). Programming the Universe: A Quantum Computer Scientist Takes On the Cosmos, Knopf, March 14.

[12] Nielsen, M. A., & Chuang, I. (2002). Quantum computation and quantum information.

[13] Maslov, D., Nam, Y., & Kim, J. (2019). An outlook for quantum computing [point of view]. Proceedings of the IEEE, 107(1), 5-10.

[14] Mc Gettrick, M., & Murphy, B. Simulation of the CCC-Not quantum gate (Vol. 61002). Technical Report NUIG-IT.

[15] Moret-Bonillo, V. (2017). Adventures in computer science: From classical bits to quantum bits. Springer.

[16] Orthey, A. C., Amorim, E. P. (2017). Asymptotic entanglement in quantum walks from delocalized initial states. Quantum Information Processing, 16(9), 224.

[17] Prieto, Alberto. [Alberto Prieto Espinosa]. (1 de abril, 2019). Computación Cuántica: puertas lógicas y algorítmos [YouTube]. Recuperado de https://www.youtube.com/watch?v=FNV0F6hdHuI

[18] Prieto, Alberto. [Alberto Prieto Espinosa]. (1 de abril, 2019). Computación Cuántica: implementaciones y arquitecturas [YouTube]. Recuperado de https://www.youtube.com/watch?v=ESAAXJXmtwY

[19] Vazirani, U. (2020). Quantum Mechanics and Quantum Computation. Recuperado de https://www.edx.org/course/quantum-mechanics-and-quantum-computation

[20] Wichert, A. M. (2013). Principles of quantum artificial intelligence. World scientific.