Escuela
Politécnica
Superior

Investigación sobre la Computación Cuántica

Grado en Ingeniería Informática

Trabajo Fin de Grado

Autor:
Sergio Claramunt Carriles
Tutor/es:
Jorge Calvo Zaragoza
Francisco José Castellanos Regalado

Mayo 2022

Universitat d'Alacant
Universidad de Alicante

Investigación sobre la Computación Cuántica

Estudio de la Computación Cuántica en los diferentes ámbitos de la informática

Autor
Sergio Claramunt Carriles

Tutor/es
Jorge Calvo Zaragoza
Departamento de Lenguajes y Sistemas Informáticos
Francisco José Castellanos Regalado
Departamento de Lenguajes y Sistemas Informáticos

Grado en Ingeniería Informática

Escuela
Politécnica
Superior

Universitat d'Alacant
Universidad de Alicante

ALICANTE, Mayo 2022

Preámbulo

Investigar sobre la computación cuántica no es algo que siempre haya deseado desde un inicio, la motivación inicial fue algo más básico, algo más relacionado con nuestra sociedad actual y con las necesidades de esta. Desde hace unos cuantos años comencé a pensar que el tiempo era un tema que me parecía algo bastante importante en todos los aspectos de la vida de una persona, quizás en mayor o menor medida y quizás más desde unas perspectivas que de otras, pero al final, consideré que el tiempo es el recurso mas importante que una persona pueda tener.

Durante toda la carrera de Ingeniería Informática, se nos ha comentado o mostrado muchos de los algoritmos o problemas que el tiempo necesario para su resolución en muchos casos es algo intratable. No solo ya dentro del ámbito de la informática sino en muchos otros ámbitos que me parecen muy importantes, como por ejemplo, la investigación de nuevas medicinas. Estos proyectos como muchos otros requieren de mucho tiempo para obtener soluciones, soluciones que quizás no son las esperadas. Esta problemática fue una de mis principales motivaciones para intentar encontrar una solución algo más práctica y general que utilizar o diseñar mejores algoritmos.

Una de las asignaturas de la carrera, Ingeniería de los Computadores, me dio paso a conocer una nueva tecnología que tenía el potencial para resolver todos estos problemas, la computación cuántica.

Como consecuencia, decidí realizar este trabajo sobre este nuevo paradigma. Durante todo este se podrán observar las características de la computación cuántica y como esta afecta tanto positiva como negativamente a los diferentes ámbitos de la informática. Además, se comentarán diferentes casos donde esta nueva tecnología podría ayudar a nuestra sociedad en este tema que tanto me importa, el tiempo.

Este trabajo, no tiene un público objetivo en concreto, cualquier persona con conocimientos en informática podría estar interesado y ser le útil este trabajo. Sin embargo, me gustaría que este trabajo fuese uno de los alicientes para que algún lector este interesado en profundizar en alguno de estos temas o en sí en este nuevo paradigma de computación.

Resumen

La computación cuántica es una nueva forma de entender la informática donde la tecnología se aprovecha de las propiedades de la mecánica cuántica. La computación cuántica hace uso de los bits cuánticos, denominados cúbits, que cuentan con las propiedades cuánticas necesarias que permiten dotar a los ordenadores de una gran velocidad de procesamiento.

Actualmente, se cuenta con los ordenadores clásicos, pero estos portan una problemática a considerar. A medida que la tecnología avanza, la velocidad de procesamiento demandada es mayor y por lo tanto se deben mejorar aspectos en el hardware de estos dispositivos. La mejora del hardware en estos dispositivos tiene un límite el cual puede cruzarse utilizando la computación cuántica con la cual, se puede conseguir satisfacer la demanda que actualmente los ordenadores clásicos les cuesta resolver o que simplemente no son capaces de resolver debido al exceso de tiempo necesario para el problema a tratar.

Con los ordenadores cuánticos se puede conseguir resolver problemas mucho más rápido que con los ordenadores clásicos debido a que los ordenadores cuánticos utilizan los cúbits que se aprovechan de las leyes cuánticas. Actualmente la computación cuántica está en proceso de investigación por lo que no es posible en este momento de la tecnología utilizarla como se utiliza la computación clásica. La ventaja principal que los ordenadores cuánticos poseen es la capacidad de procesamiento, esto conlleva a reducir el tiempo de espera para la resolución de las ejecuciones de algoritmos en los ordenadores cuánticos. Sin embargo, aún es necesario seguir investigando para mejorar las infraestructuras tan exigentes que son necesarias para tener un ordenador cuántico, ya que para contar con un ordenador cuántico es necesario tener unas condiciones en cuanto a temperatura y aislamiento especiales para que el sistema funcione correctamente.

Por otra parte, una vez pasadas las barreras que actualmente la computación cuántica tiene debido al desconocimiento de esta nueva tecnología, muchas empresas podrán nutrirse del potencial que esta puede traer. Empresas de investigación, matemáticos, físicos, químicos, economía, farmacéutica, seguridad informática, inteligencia artificial, en general, todas las disciplinas y labores que conlleven el uso de gran cantidad de datos y/o resolución de algoritmos de complejidad elevada, podrán aprovecharse en gran medida de los beneficios de la computación cuántica.

En este trabajo se pretende estudiar este nuevo paradigma, con el objetivo de presentar lo que esta nueva tecnología puede traer. Además, se comentarán diferentes ámbitos de la informática donde estos tendrán ventajas y desventajas al contar con la computación cuántica. Sin embargo, actualmente estos efectos sobre los diferentes ámbitos de la informática únicamente son teóricos, se necesita de más tiempo de investigación para poder contar con una computación cuántica cada vez más real.

Agradecimientos

Quisiera transmitir mi mas sincero agradecimiento a todos aquellos que me han ayudado y apoyado en este largo e intenso proyecto.

En primer lugar me gustaría agradecer este trabajo a mis tutores Jorge Calvo Zaragoza y Francisco José Castellanos Regalado quienes me han ayudado a poder transmitir claramente todos los conceptos y todos los planteamientos que deseaba expresar.

También me gustaría destacar la ayuda y el apoyo de todos mis seres queridos por estar ahí escuchando e intentando entender algunos conceptos que con dificultad se pueden hacer comprender. Además, también me gustaría puntualizar la intervención de muchos ellos para mejorar notoriamente la comprensión y el entendimiento del trabajo.

Por último, pero no menos importante, me gustaría agradecer a la Universidad de Alicante por nutrirme sobre, más allá de conocimientos, técnicas y estrategias para poder resolver cualquier problemática que pueda surgir en el futuro.

A todos ellos, muchas gracias, sin vuestras aportaciones el camino habría sido más complicado.

Índice general

Índice de figuras

Índice de tablas

Índice de Códigos

1 Introducción

Desde 1980, la idea de realizar los cálculos de forma cuántica ha tenido mucho interés debido a la demanda de procesamiento de datos que tenemos actualmente. Con el paso del tiempo, se ha ido investigando y desarrollando la tecnología en torno a la computación cuántica hasta tal punto que se ha conseguido construir infraestructuras capaces de soportar dicha computación logrando unos resultados asombrosos en cuanto al coste temporal con respecto la computación clásica.

Este nuevo paradigma de computación salió a la luz debido a los grandes beneficios que nos podría aportar, pudiendo mejorar en gran medida los tiempos y la seguridad de los tratamientos de datos que podría ser de gran utilidad para trabajos de investigación, cálculos complejos, descubrimientos sobre nuevos medicamentos, inteligencia artificial, economía, empresas, ciberseguridad, etc. La computación cuántica puede marcar un punto de inflexión para muchos ámbitos laborales y sobre todo para muchos campos de la informática.

La computación cuántica supone un cambio conceptual en la manera de entender la informática. Adquirir los conocimientos necesarios para tratar con este nuevo método será mucho más laborioso que con la computación actual debido al uso de la mecánica cuántica. Aunque esto no será un problema ya que actualmente esta tecnología no está destinada para cualquier uso ni para cualquier ámbito laboral, sino más bien está destinado para un ámbito donde impere la necesidad de cálculos, procesos y algoritmos altamente complejos.

La computación cuántica puede ser la clave para obtener una mayor velocidad de procesamiento. Lo que con la tecnología actual podríamos tardar años en realizar, con este nuevo paradigma se podría reducir el coste temporal a unas pocas horas, pudiendo ser una revolución y un cambio en nuestra forma de vida.

1.1 Objetivos

A continuación se describen los diferentes objetivos planteados para desarrollar este trabajo. Cabe destacar que este proyecto es principalmente un trabajo de recopilación e investigación sobre un tema candente el cual puede ser de gran utilidad para entender el cambio tan grande que la computación cuántica puede llegar a realizar en nuestra tecnología actual.

- Recopilar y organizar en un mismo documento la información más relevante sobre la computación cuántica.

- Analizar las oportunidades que podrían abrirse con el desarrollo de esta tecnología.
- Estudiar y adquirir conocimientos sobre un nuevo paradigma de la informática que puede suponer una mejora drástica en muchos aspectos de la informática: la computación cuántica.
- Comprender los beneficios y los problemas que la computación cuántica puede conllevar, tanto para la informática como para las empresas y por ende para la sociedad.
- Hacer entender el gran potencial que esta por venir y para el que deberemos estar preparados.
- Estudiar y comprender sobre un campo de la informática que apenas es estudiado durante la formación universitaria para adquirir conocimientos y capacidades sobre la materia en cuestión.

1.2 Motivación

La computación cuántica está creciendo poco a poco y su consolidación podría convertirse en una revolución tecnológica ya que muchas de las estrategias e infraestructuras que utilizamos actualmente en los distintos ámbitos de la informática no servirían.

Hay mucha información sobre computación cuántica que no está convenientemente organizada. Este trabajo pretende unificar la información más relevante de esta temática con el objetivo de facilitar en un futuro la obtención de unos conocimientos mínimos sobre el funcionamiento de la computación cuántica, así como las posibles aplicaciones e inconvenientes que puedan conllevar la implantación de este nuevo paradigma.

Todo esto con el objetivo de intentar anticiparnos a un gran cambio que podría estar por venir. Podría ser una revolución tecnológica de gran magnitud por la cantidad de beneficios que esta podría traer a una sociedad donde las comunicaciones y el tratamiento de grandes cantidades de datos son cruciales para nuestras vidas.

1.3 Estructura del trabajo

A continuación se describe la estructura del proyecto donde se puede observar las diferentes partes de este. Además, se realizará una breve explicación sobre el contenido de cada capítulo.

- **Capitulo 2 - Computación cuántica:** Se describen aspectos básicos sobre la computación cuántica, las novedades en cuanto a la lógica de esta nueva computación y las ventajas y desventajas de esta.
- **Capítulo 3 - Computación cuántica en la seguridad informática:** Se tratan

aspectos de criptografía cuántica y los problemas que podría acarrear junto con otros temas como la computación cuántica en las criptomonedas.

- **Capítulo 4 - Computación cuántica en redes informáticas:** Se introduce la idea de un internet cuántico junto con un experimento demostrable que podría dar luz a este novedoso concepto.

- **Capítulo 5 - Computación cuántica en lenguajes de programación:** Se introducen algunos lenguajes actuales que permiten comunicarse con un ordenador cuántico. Además se aporta una idea de lo que sería el proceso de comunicación entre el lenguaje de alto nivel con la propia unidad básica de la computación cuántica.

- **Capítulo 6 - Computación cuántica en la algorítmia:** Se analizan diferentes algoritmos que tiene solución en el ámbito de la computación cuántica intentando siempre realizar una comparación con una posible solución en computación clásica para observar la mejora que la computación cuántica tiene sobre la computación clásica.

- **Capítulo 7 - Computación cuántica en inteligencia artificial:** Se analizan los cambios en la estructura del funcionamiento de la inteligencia artificial y además los grandes beneficios que la computación cuántica otorga a esta.

- **Capítulo 8 - Aplicaciones con computación cuántica:** Se analizan algunos experimentos para demostrar actualmente el potencial de la computación cuántica y se realizan dos circuitos cuánticos para ver diferentes propiedades que se comentan a lo largo del trabajo.

- **Capítulo 9 - Computación cuántica en la actualidad:** Se recoge la información de un breve período de tiempo para dar a entender la evolución y las mejoras que se están teniendo en la computación cuántica

- **Capítulo 10 - Conclusión:** Una breve conclusión sobre aspectos generales de la computación cuántica comentados en profundidad en todo el trabajo y un análisis sobre la computación cuántica y sobre la influencia que podría tener esta sobre la tecnología actual.

2 Computación cuántica

Para comprender bien lo que es la computación cuántica se va a comenzar explicando brevemente lo que es la mecánica cuántica.

La **mecánica cuántica** es una parte de la física que estudia la materia a nivel molecular, atómico y subatómico y nos permite estudiar el comportamiento de esta materia con dimensiones muy pequeñas. La teoría cuántica solo permite normalmente cálculos probabilísticos o estadísticos de las características observadas de las partículas elementales.

La mecánica cuántica contiene un conjunto de **leyes cuánticas** que define este comportamiento de la materia de dimensiones muy pequeñas. Las leyes cuánticas pueden proporcionar propiedades como *estado cuántico*, *observable*, *medición*, *superposición*, *entrelazado*, *decoherencia*, *dualidad onda-partícula*, entre otras. Algunas de estas propiedades se explicarán en profundidad a lo largo del proyecto

Los **ordenadores cuánticos** sencillamente aprovechan la mecánica cuántica para aumentar notoriamente la velocidad de procesamiento actual incluso mejorando a los supercomputadores con los que hoy en día contamos.

Por otra parte, los ordenadores cuánticos no cuentan con los bits convencionales con los que contamos actualmente en la computación clásica (cuyo valor puede ser 1 ó 0). Estos utilizan los bits cuánticos o **cúbits**, los cuales son generados gracias a esas leyes cuánticas, generando partículas subatómicas como electrones o fotones. Gracias a los cúbits podemos ser capaces de obtener mucha más velocidad de procesamiento que con la misma cantidad de bits de la computación clásica ya que los cúbits se aprovechan de las propiedades de las leyes cuánticas. Mas adelante se explica con detalle como surge este aumento de velocidad de procesamiento gracias a los cúbits.

La computación clásica actualmente puede resolver algoritmos de gran complejidad en un tiempo muy elevado, pero con la computación cuántica se podrían resolver estos mismos algoritmos de complejidades elevadas con un menor tiempo de ejecución. Muchos de los problemas que surgen debido a estas altas complejidades en la computación clásica se resolverían con la computación cuántica.

Aún con las muchas diferencias que la computación cuántica y la clásica poseen, hay aspectos básicos que se asemejan entre sí, como el uso de la **máquina de *Turing***, que a continuación se va a comentar, o el uso de las **puertas lógicas** que pueden verse detalladamente más adelante en el apartado 2.3.

La máquina de *Turing* es el modelo sobre el cual se construye la lógica y la arquitectura de los ordenadores actuales ya que es capaz de simular la lógica de cualquier algoritmo de un ordenador. La máquina de *Turing* consta de una cinta que actúa como memoria, un cabezal con la capacidad de leer, escribir y moverse por la cinta de derecha a izquierda, un registro del estado y una tabla de instrucciones.

El funcionamiento de esta máquina, como se puede observar en la figura 2.1 se basa en que el cabezal lee los símbolos que va obteniendo de la cinta y dependiendo del símbolo, la reacción del cabezal será escribir en una celda de la cinta o moverse entre las celdas de la cinta. El resultado del algoritmo se escribe en la misma cinta donde estaban los símbolos entrantes

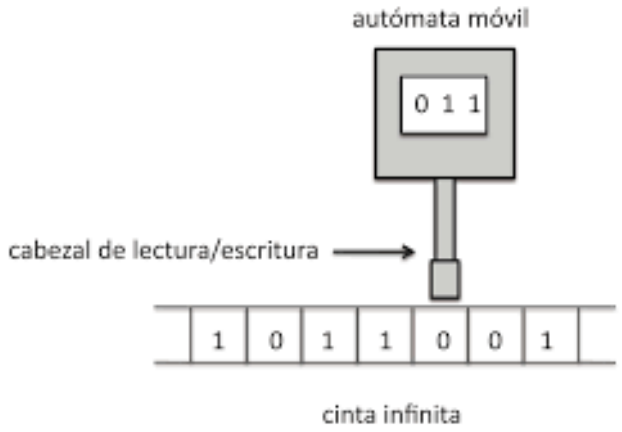


Figura 2.1: Máquina de *Turing* clásica. Obtenida de [71].

Si esta máquina de *Turing* corresponde al funcionamiento de un ordenador clásico, los ordenadores cuánticos cuentan con su propia máquina de *Turing*: la **máquina de *Turing* cuántica**.

2.1 Máquina de Turing Cuántica

En 1985, Deutsch mostró el diseño de la primera máquina cuántica teniendo como base la actual máquina de *Turing*.

La maquina de *Turing* cuántica es muy similar a la clásica, y esta compuesta por estos elementos:

- Cinta de memoria infinita
- Procesador finito

- Cabezal

La **cinta de memoria infinita** es muy similar a la de la máquina de *Turing* clásica, lo único es que los elementos de la cinta de la máquina cuántica no son bits, sino que son cúbits.

El **procesador finito** tiene el conjunto de instrucciones que se van aplicando sobre cada elemento de la cinta señalado por el cabezal.

El **cabezal** únicamente lee y escribe en la cinta, pudiéndose mover por esta de izquierda a derecha buscando la posición o el elemento.

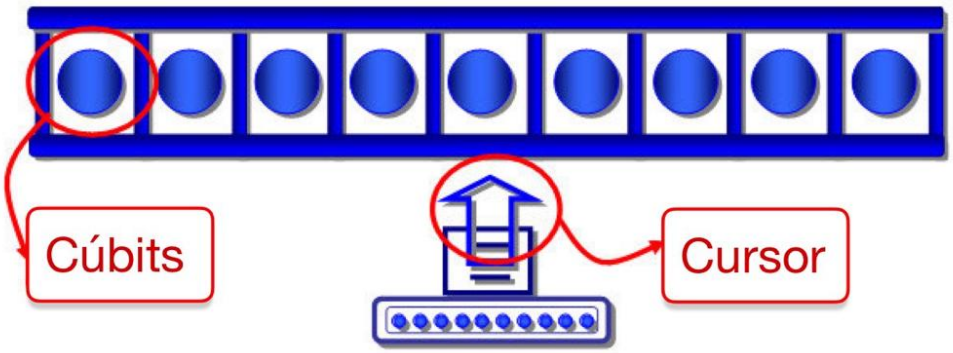


Figura 2.2: Máquina de *Turing* cuántica. Obtenida de [90].

2.2 Cúbit

El cúbit (*qubit*) es la unidad básica de información en la computación cuántica como lo es el bit binario para la computación clásica.

Los cúbits tienen varias características que los definen y que hacen que la computación cuántica sea tan potente: la **superposición de estados**, y el **entrelazamiento cuántico**.

2.2.1 Superposición de estados

La superposición da la capacidad de estar simultáneamente en múltiples estados lo que permite a los cúbits representar muchas combinaciones de unos y ceros al mismo tiempo. Controlando y variando estos estados podemos utilizar varios cúbits para procesar gran cantidad de información.

La superposición da a los equipos cuánticos la capacidad de realizar cálculos más complejos rápidamente, es decir, permite que los algoritmos cuánticos procesen la información en un breve periodo de tiempo en comparación a los sistemas clásicos.

Mientras que un bit puede estar únicamente en uno de sus estados (1 ó 0), un cúbit puede estar en una **superposición de ambos**, una mezcla de los dos estados.

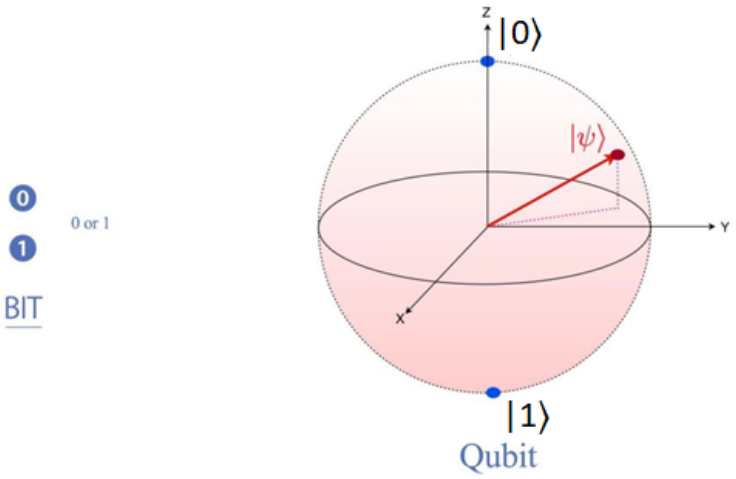


Figura 2.3: Esfera de *Bloch*. Obtenida de [69].

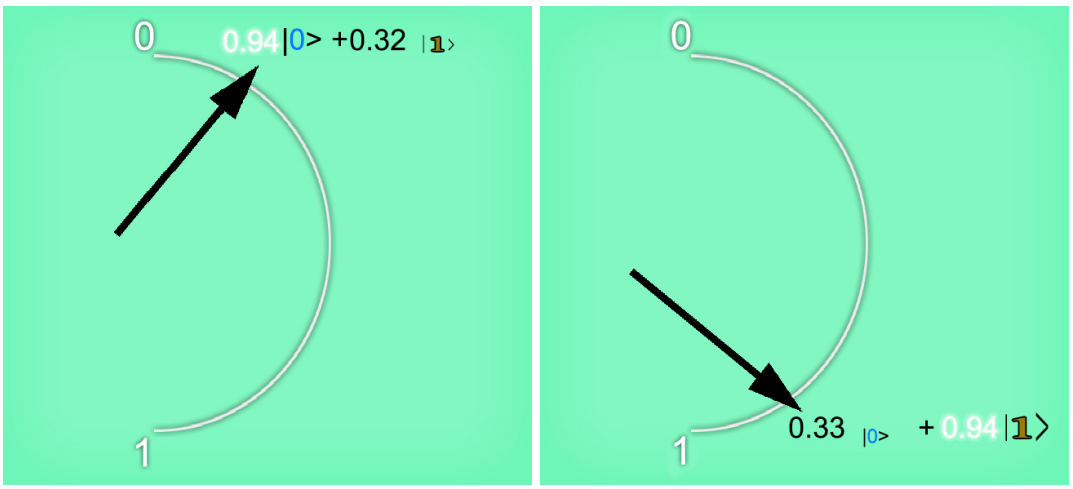
Como podemos observar en la Figura 2.3, el bit puede variar entre 0 o 1 y solo puede tener uno de esos dos estados.

Por otra parte, el cúbit se representa como una esfera donde en los dos polos están situados el 1 y el 0, y a partir de estos podemos tomar cualquier valor que será determinada por ψ, de forma que si ψ esta más cerca de 0, mas relevante será el estado de 0 mientras que más cerca este de 1 más relevante será el estado de 1, como se refleja en la figura 2.4.

Al ser una esfera también podemos acercarnos al 1 por el lado negativo del eje de coordenadas, por lo que también puede tomar estado negativo. No solo eso, sino que los coeficientes de las mezclas pueden hacerse con números complejos, es decir, que un solo cúbit puede configurarse de infinitas formas.

Por otra parte, como la superposición es arbitraria, es decir, no sabemos que estados se van a tomar, hay veces que las probabilidades de que sea 0 o que sea 1 sean prácticamente iguales (p. ej. 0: 58%, 1: 42%) por lo tanto el resultado será impredecible, no sabremos qué valor se tomará y esta arbitrariedad no se debe a un factor humano, sino que se debe a las propias leyes cuánticas que determinan que el cúbit debe comportarse así.

Obtener un resultado impredecible puede significar que el resultado puede ser erróneo o no. Esto es algo habitual ya que se esta tratando con probabilidades. Sin embargo, existen



(a) Más probabilidad de 0. **(b)** Más probabilidad de 1.

Figura 2.4: Elección probabilística del estado del cúbit. Imagen basada en [18].

propiedades cuánticas que más adelante se comentarán como el **entrelazamiento cuántico** o la **corrección de errores por la decoherencia cuántica** que explican que se puede "forzar" y "asegurar" un resultado.

De forma resumida para el cúbit, cuanto más cerca este ψ de 0 más importante será el estado 0 y cuanto más cerca este de 1 más importante será el estado 1. Hasta ahora, la medición para determinar el estado ha sido mediante el eje 'z' (véase la Figura 2.3),donde ψ variaba entre 0 y 1. Pero también se puede hacer la medición sobre el eje 'x' (véase la Figura 2.3), siendo el resultado más probable derecho o más probable izquierdo, es decir, hay varias formas de **medir** los posibles ejes de los cúbits y con la posibilidad de realizar combinaciones de mediciones, es por esto que los cúbits pueden almacenar grandes cantidades de información ya que pueden tener prácticamente infinitos estados.

En la figura 2.5 se puede observar que inicialmente el estado del cúbit es 1 y 0 al mismo tiempo, teniendo cada valor una probabilidad en concreto. Esto ocurre cuando no se aplica la "medición", que el cúbit podría tomar cualquiera de los dos valores.

Mientras tanto, en la figura 2.6, se realiza la "medición", es decir, que se observa el valor del cúbit y por tanto toma un valor fijo.

Gracias a estas características, si la computación clásica necesita una cantidad inviable de tiempo para encontrar los factores primos de un numero de 2048 bits, la computación cuántica con los cúbits podría obtener el resultado en unas pocas horas. Esto es gracias a que la cantidad de información que puede representar un cúbit es enorme. La información que habría en 300 cúbits no podría ser representada ni con más de 2^{300} bits clásicos.

Es decir, necesitamos un bit para representar un 1 o un 0, pero en un cúbit podemos representar un 1 y un 0 al mismo tiempo, por ende, con 2 bits tendríamos 2 resultados (1 o

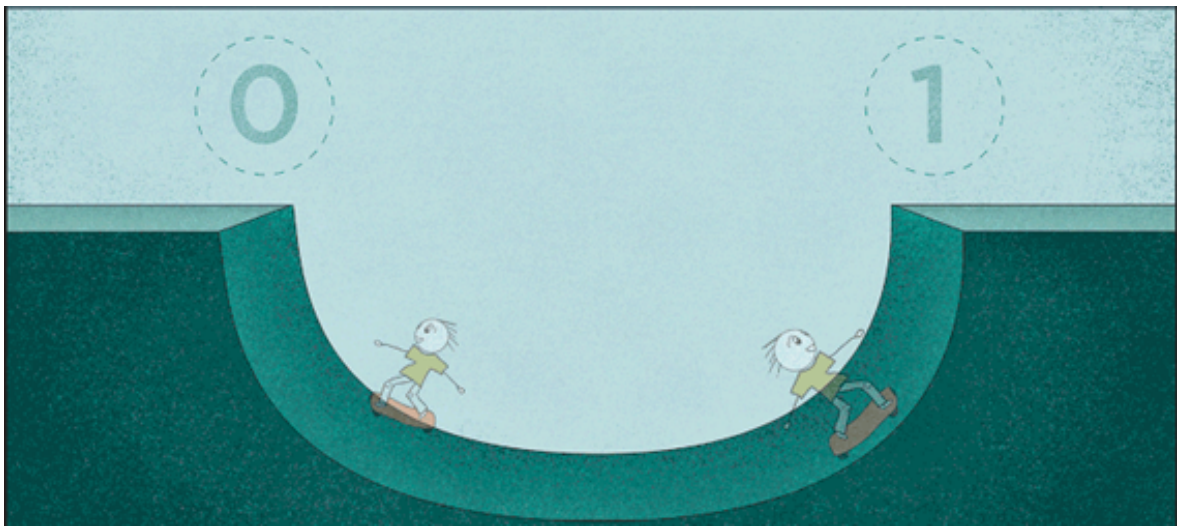


Figura 2.5: cúbit sin realizar medición. Obtenida de [25].

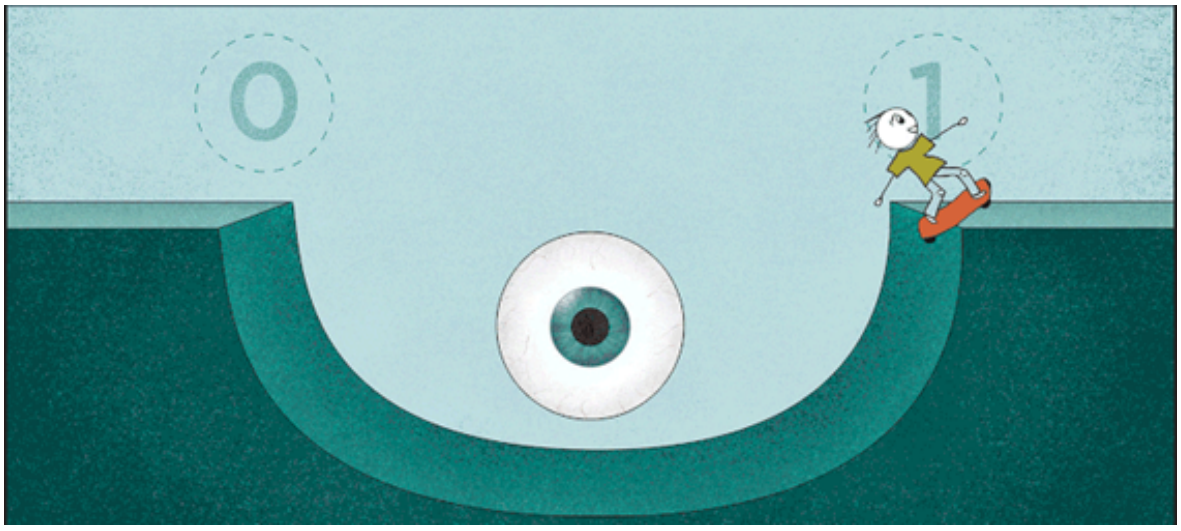


Figura 2.6: cúbit realizando medición. Obtenida de [25].

0 del primer bit y 1 o 0 del segundo bit) pero con 2 cúbits tendríamos 4 resultados (1 y 0 del primer cúbit y 1 y 0 del segundo cúbit), por lo tanto, podemos observar como el cúbit crece a la proporción de 2^n, lo que quiere decir que, por ejemplo, para 20 bits obtendríamos 20 resultados, pero para 20 cúbits obtendríamos 2^{20} resultados, es decir, 1.048.576 resultados.

Además, cada proceso se realiza de forma independiente al resto, por lo que se pueden resolver **operaciones en paralelo**.

2.2.2 Entrelazamiento cuántico

El entrelazamiento cuántico es un recurso de la mecánica cuántica que es necesario generar en los cúbits para obtener el máximo efecto en la computación cuántica. Permite que los cúbits puedan interactuar entre sí independientemente de la distancia entre estos, pero con la peculiaridad de que deben permanecer aislados.

Como se puede observar en la figura 2.7, el entrelazamiento se da cuando, partículas que han interactuado entre sí, retienen un tipo de conexión y pueden entrelazarse. Este proceso se conoce como **correlación**. Dos partículas con entrelazamiento giran en sentidos opuestos por lo tanto si conocemos el sentido del giro de una partícula conoceremos el de la otra y esto, evidentemente, es aplicable en los cúbits.

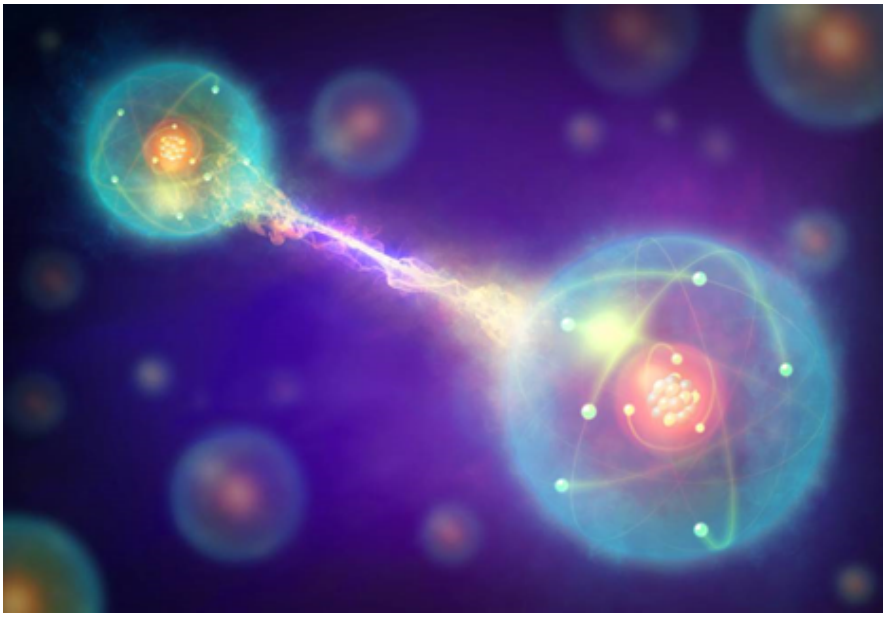


Figura 2.7: Entrelazamiento cuántico. Obtenida de [66].

Esta propiedad solo se da cuando el cúbit está en superposición. Por ejemplo, supongamos que tenemos dos monedas, donde la cara representa el valor 1 y el reverso el valor 0. Cuando las lanzamos al aire y comienzan a girar, las monedas están superpuestas, es decir, tienen a la vez el valor 1 y el valor 0. Además, suponiendo que puedan tener un entrelazado cuántico, una giraría en un sentido y la otra en otro sentido, pero las dos seguirían girando. En el momento en el que las monedas caigan, dejarían de girar y por tanto se obtendría un resultado, por lo que no habría superposición y por tanto tampoco habría entrelazado, pero sabríamos el valor de las dos por haber estado entrelazadas. De la misma forma sucedería con los cúbits.

2.2.2.1 Teletransportación cuántica

El teletransporte cuántico se fundamenta en una hipótesis, descrita en [40] y realizada por *Albert Einstein*, *Boris Podolsky* y *Nathan Rosen* en 1935. Con esta característica dos partículas las podemos separar en el espacio y aun seguirían compartiendo sus propiedades.

Desde 1998 se realizaron diversos experimentos que lograron el teletransporte cuántico utilizando fotones, átomos y sistemas más complejos. En un primer momento se demostró este teletransporte a distancias cortas, pero se fue aumentando la distancia gradualmente hasta obtener el récord que actualmente existe, un teletransporte de fotones a 1400 kilómetros de distancia desde la Tierra [33].

Esta misma idea con el cúbit no es tan sencillo de aplicar porque el teletransporte cuántico no sirve para transmitir datos instantáneamente o a velocidades elevadas como la de la luz. Esto ocurre cuando un cúbit esta entrelazado a distancias muy largas y se debe establecer una comunicación para obtener información adicional de las mediciones realizadas por un cúbit y por el otro. Como solución a este problema, es necesario enviar unos cuantos bits clásicos más, es decir, que por cada cúbit teletransportado se deben transmitir dos bits clásicos. Estos bits clásicos solo pueden transmitirse por las vías clásicas que ya actualmente utilizamos con normalidad, por lo tanto, la velocidad a alcanzar no ha mejorado notoriamente.

Pero esta limitación solo está contemplada en los cúbits. A medida que se va investigando y experimentando surgen nuevas formas de transmisión, como la transmisión de *qutrits*, o unidades de transmisión tridimensional, capaces de tomar tres valores (0, 1 y 2) o como la transmisión de *ququart*, capaces de tomar cuatro valores (0, 1, 2 y 3). Gracias a estas investigaciones de unidades de transmisión como cúbit, *qutrit*, *ququart*, etc… pueden abrir las puertas a un futuro para las redes de computación cuántica, lo que aumentaría la capacidad de información en la transmisión y mayor resistencia al ruido, entre otros beneficios.

Por lo que pasar de un cúbit a un *qutrit* y de este a un *ququart* y así sucesivamente, será lo que en un futuro nos permitirá sentar las bases de las **redes de computación cuántica**. El cambio de una unidad de transmisión a otra implica únicamente tener un nuevo estado en los cúbits. Por ejemplo, si a un cúbit le añadimos un nuevo estado, obtenemos un *qutrit*. El tener un estado más implica un aumento del procesamiento de los cúbits.

2.2.2.2 Interferencia cuántica

Hasta ahora se ha explicado generalmente que los cúbits pueden tomar 2 valores a la vez, pero esto es de poco valor en un ordenador cuántico ya que se están dando dos respuestas para un problema con una única respuesta correcta. Es aquí donde sale a la luz otro fenómeno cuántico, la interferencia cuántica.

En física, las ondas y las partículas son distintas pero *Albert Einstein* en 1905 introdujo la dualidad onda-partícula. Aunque ya se tenían casos, como en el de la luz, donde no era una onda o una partícula sino era ambas cosas. Esto mismo ocurre con los cúbits y es por eso que

dos cúbits tienen su propia onda y al interaccionar una con la otra pueden ocurrir dos cosas: que la interacción dé como resultado una **interferencia destructiva**, o una **interferencia constructiva**.

Vamos a explicar este fenómeno con un ejemplo. Imaginemos que arrojamos una piedra sobre un lago y esta genera una onda debido al impacto. Estas ondas en el agua tienen forma de círculos concéntricos que pueden, al mismo tiempo, atravesar dos puentes vecinos en el propio lago. Si suponemos que un puente se representa por el 0 y el otro por el 1, parte de la onda es 0 y 1 simultáneamente. Para un ordenador cuántico dar dos respuestas a un problema no aporta una solución fiable es por tanto que interviene el fenómeno de la interferencia cuántica. Con esta, si la ondas ondas se cancelan (**interferencia destructiva**) el resultado es 0 y si las ondas se suman(**interferencia constructiva**) el resultado es 1. Estas interferencias se pueden observar en la figura 2.8.

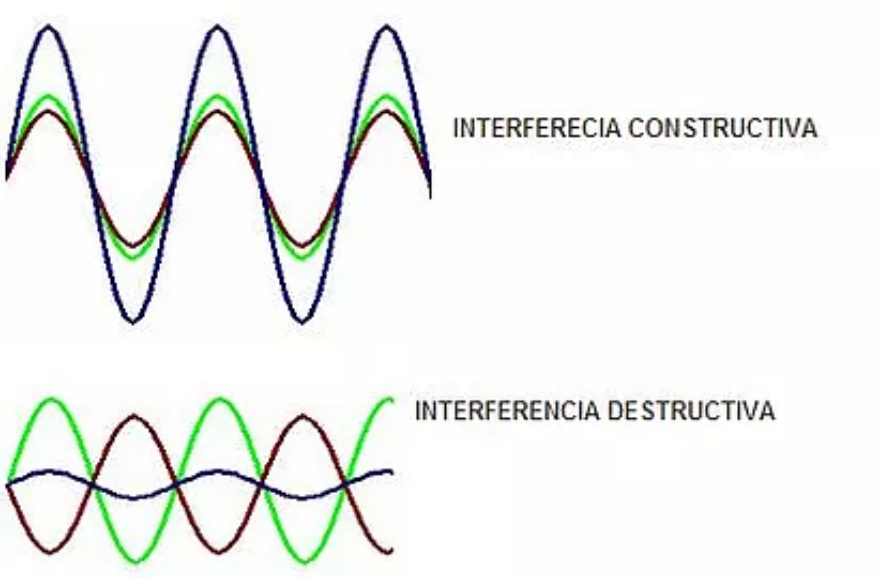


Figura 2.8: Interferencia cuántica (la curva azul es el resultado de la suma de las otras dos curvas). Obtenida de [51].

2.3 Puertas Cuánticas

Las puertas lógicas actuales son dispositivos electrónicos que poseen funciones capaces de realizar cálculos que operan a nivel de bit. En la figura 2.9 se pueden observar las diferentes puertas lógicas con las que contamos, donde la parte izquierda de cada puerta lógica son los bits entrantes y la parte derecha el bit resultante de la operación.

El resultado de las operaciones de cada puerta lógica se determina según su tabla de la verdad correspondiente.

Por ejemplo, para la puerta "AND" de la figura, las entradas de esta son 2 bits, "A" y "B" dando como resultado el bit "A AND B". Dependiendo de si "A" o "B" son 1 o 0, el resultado será uno u otro, como se puede observar en la tabla 2.1.

Tabla 2.1: Tabla de la verdad AND.

A	B	A AND B
0	0	0
0	1	0
1	0	0
1	1	1

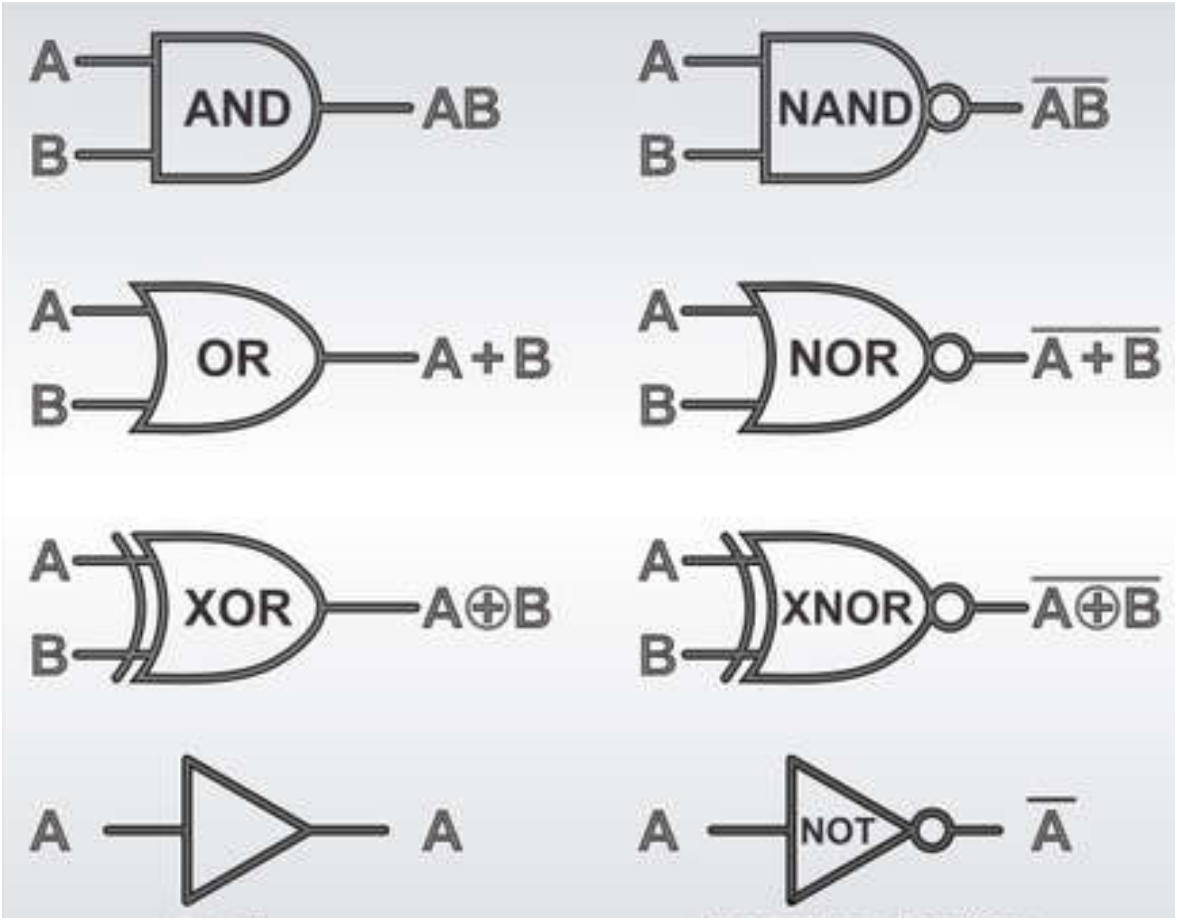


Figura 2.9: Puertas lógicas clásicas. Obtenida de [91].

Análogamente a las puertas lógicas descritas previamente, las **puertas cuánticas** son circuitos cuánticos básicos que operan sobre cúbits y evidentemente sobre ordenadores cuánticos. Estas se suelen representar como matrices. Una puerta que opera sobre n cúbits queda

representada por una matriz unitaria de $2^n \times 2^n$. Nótese que el número de cúbits en la entrada debe coincidir con el número de cúbits en la salida.

Una característica muy importante de las puertas cuánticas, es que son **reversibles**, es decir, siempre va a ser posible obtener las entradas a partir de las salidas, lo cual no ocurre generalmente en las puertas lógicas convencionales (en la sección 8.4.2 se realiza una demostración sobre esta característica). Las puertas cuánticas más comunes operan en espacios de uno o dos cúbits de la misma forma que en las puertas clásicas operan en uno o dos bits.

Los estados cuánticos se representan mediante "$|binario\rangle$" (*kets*). De forma que el estado "$|0\rangle$" o el ket 0 equivale a la ecuación 2.1 y el estado "$|1\rangle$" o el ket 1 equivale a la ecuación 2.2.

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{2.1}$$

$$|1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{2.2}$$

El motivo por el cual el estado de un cúbit se representa por una matriz que contiene dos valores es por el hecho de que estos, como se ha explicado previamente, pueden ser 1 y 0 a la vez. A medida que se van aplicando las diferentes puertas cuánticas, estos cúbits varían su resultado según la funcionalidad de la puerta cuántica que se le aplique.

La combinación de diferentes estados cuánticos se realiza mediante el Producto de *Kronecker*[1], representado con el símbolo "$\otimes$". De forma que si combinamos un estado a y un estado b obtendríamos el resultado en la ecuación 2.3. Esta combinación es necesaria cuando se utilizan puertas cuánticas donde se requieren dos o mas cúbits. Puertas como **CNOT** o **Toffoli**, que se explicarán mas adelante, utilizan varios cúbits y será necesario realizar esta combinación para poder utilizar dichas puertas.

$$|ab\rangle = |a\rangle \otimes |b\rangle, \tag{2.3}$$

siendo $a = |0\rangle \; o \; |1\rangle, \; y \; b = |0\rangle \; o \; |1\rangle$.

Para aclarar esta operación se va a realizar un ejemplo con el objetivo de entender posteriormente las ecuaciones que se van a utilizar. Imaginemos que $|a\rangle = 0$ y que $|b\rangle = 1$, el resultado quedaría reflejado en la ecuación 2.4.

Únicamente destacar que en la ecuación 2.4, los subíndices de los números son simplemente orientativos, para conocer con detalle los pasos en las operaciones que se hacen en el Producto

[1] Operación sobre dos matrices donde cada elemento de una matriz se multiplica por todos los elementos de la otra matriz generando una matriz nueva con todos los resultados de las multiplicaciones.

de *Kronecker*.

$$|ab\rangle = |01\rangle = |0\rangle \otimes |1\rangle = \begin{bmatrix} 1_1 \\ 0_2 \end{bmatrix} \otimes \begin{bmatrix} 0_3 \\ 1_4 \end{bmatrix} = \begin{bmatrix} 1_1 \cdot 0_3 \\ 1_1 \cdot 1_4 \\ 0_2 \cdot 0_3 \\ 0_2 \cdot 1_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \tag{2.4}$$

A continuación, se puede observar en la figura 2.10 algunas de las puertas cuánticas con las que contamos en la computación cuántica.

Puerta Cuántica	Qubits	Ecuación
Puerta Hadammard (H)	1	$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$
Puerta X	1	$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
Puerta Y	1	$Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$
Puerta Z	1	$Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$
Puerta SWAP	2	$SWAP = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
Puerta CNOT	2	$CNOT = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$
Puerta Toffoli	3	$Toffoli = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$

Figura 2.10: Puertas Cuánticas más importantes.

Seguidamente se comentarán las puertas cuánticas vistas en la figura 2.10, pero su uso sobre circuitería cuántica puede verse detalladamente en la sección 8.4.

2.3.1 Puerta de Hadamard

La puerta de *Hadamard* trabaja únicamente con un cúbit y se encarga de realizar la superposición en los cúbits entrantes en dicha puerta. El circuito de la puerta de *Hadamard* se representa por la figura 2.11 y por la ecuación 2.5.

$$H(q) = q \cdot \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \tag{2.5}$$

siendo $q = |0\rangle$ o $|1\rangle$.

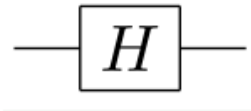


Figura 2.11: Puerta cuántica H. Obtenida de [48].

2.3.2 Puerta X

La puerta X trabaja únicamente con un cúbit. Esta puerta realiza la misma acción que una puerta NOT en la computación clásica, es decir, se realiza el cambio de estado de $|0\rangle$ a $|1\rangle$ y de $|1\rangle$ a $|0\rangle$. El circuito de la puerta X se representa por la figura 2.12 y por la ecuación 2.6.

$$X(q) = q \cdot \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \tag{2.6}$$

siendo $q = |0\rangle$ o $|1\rangle$.

Figura 2.12: Puerta cuántica X. Obtenida de [48].

2.3.3 Puerta Y

La puerta Y trabaja únicamente con un cúbit. Se realiza el cambio de estado de $|0\rangle$ a $i|1\rangle$ y de $|1\rangle$ a $-i|0\rangle$. El circuito de la puerta Y se representa por la figura 2.13 y por la ecuación

2.7. La "i" no afecta al estado resultante, unicamente a la fase en la que se encuentra el cúbit dentro de la esfera de Bloch, como se puede recordar en la figura 2.3. Si el cúbit de entrada es $|0\rangle$ y aplicamos la puerta Y, obtendremos $|1\rangle$ junto con la fase $\frac{\pi}{2}$. Si el cúbit de entrada es $|1\rangle$ y aplicamos la puerta Y, obtendremos $|0\rangle$ junto con la fase $\frac{3\pi}{2}$.

$$Y(q) = q \cdot \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, \tag{2.7}$$

siendo $q = |0\rangle$ o $|1\rangle$.

Figura 2.13: Puerta cuántica Y. Obtenida de [48].

2.3.4 Puerta Z

La puerta Z trabaja únicamente con un cúbit. Esta puerta es un caso especial de puerta de cambio de fase ya que deja el estado $|0\rangle$ sin cambios y el estado $|1\rangle$ a $|\text{-}1\rangle$. El circuito de la puerta Z se representa por la figura 2.14 y por la ecuación 2.8.

$$Z(q) = q \cdot \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \tag{2.8}$$

siendo $q = |0\rangle$ o $|1\rangle$.

Figura 2.14: Puerta cuántica Z. Obtenida de [48].

2.3.5 Puerta SWAP

La puerta SWAP trabaja con dos cúbits. Esta puerta intercambia el valor de un cúbit por el valor del otro cúbit. El circuito de la puerta SWAP se representa por la figura 2.15 y por la ecuación 2.9.

$$SWAP(q_1, q_2) = |q_1\rangle \otimes |q_2\rangle = |q_1 q_2\rangle \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{2.9}$$

siendo $q_1 = |0\rangle \ o \ |1\rangle, \ y \ q_2 = |0\rangle \ o \ |1\rangle$.

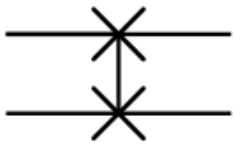


Figura 2.15: Puerta cuántica SWAP. Obtenida de [48].

2.3.6 Puerta CNOT

La puerta CNOT o "NOT controlada" es una puerta controlada y estas se caracterizan por tener algunos cúbits que son de control, es decir, que dependiendo del estado de los cúbits de control el resultado será uno u otra.

CNOT trabaja sobre 2 cúbits, donde un cúbit es de control y el otro donde se realiza la operación, solo si el cúbit de control es apto. Al aplicar la puerta CNOT sobre 2 cúbits, se determina cual será el cúbit de control. Posteriormente, al ejecutar la puerta, si el cúbit de control tiene el estado $|1\rangle$ entonces el otro cúbit cambiará su estado al opuesto al cual estaba previamente, es decir, si tenia el estado $|1\rangle$ pasará a tener el estado $|0\rangle$ y viceversa. El circuito de la puerta CNOT se representa por la figura 2.16 y por la ecuación 2.10.

$$CNOT(q_c, q_1) = |q_c\rangle \otimes |q_1\rangle = |q_c q_1\rangle \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \tag{2.10}$$

siendo $q_c = |0\rangle \ o \ |1\rangle, \ y \ q_1 = |0\rangle \ o \ |1\rangle$.

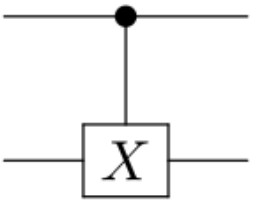


Figura 2.16: Puerta cuántica CNOT o CX. Obtenida de [48].

A continuación, se realiza una demostración de como se consigue obtener la puerta CNOT a través de diferentes operaciones. En esta se puede observar el tipo de operaciones con el que las puertas cuánticas trabajan.

2.3.6.1 Demostración de la puerta CNOT

En este apartado vamos a demostrar como se ha conseguido llegar hasta la creación de la puerta CNOT con el objetivo de dar a conocer el tipo de operaciones que se realizan utilizando puertas cuánticas.

Inicialmente, recordemos que contamos con los estados: $|0\rangle$ y $|1\rangle$. Estos se representan por matrices como se observa en la ecuación 2.11

$$\begin{aligned} |0\rangle &= \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\ |1\rangle &= \begin{bmatrix} 0 \\ 1 \end{bmatrix} \end{aligned} \tag{2.11}$$

Seguidamente se debe realizar la combinación de los estados posibles que pueden haber con 2 cúbits. Es necesario realizar la combinación de todos los estados posibles ya que la puerta CNOT debe contemplar todas las posibles entradas y además las entradas deben ser con 2 cúbits. En la ecuación 2.12 se puede observar las operaciones para obtener cada combinación de estados.

$$\begin{aligned} |0\rangle \otimes |0\rangle &= \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = |00\rangle \\ |0\rangle \otimes |1\rangle &= \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} = |01\rangle \\ |1\rangle \otimes |0\rangle &= \begin{bmatrix} 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} = |10\rangle \\ |1\rangle \otimes |1\rangle &= \begin{bmatrix} 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = |11\rangle \end{aligned} \tag{2.12}$$

Una vez tenemos estas combinaciones, para montar la matriz CNOT debemos realizar XOR ($\oplus$) sobre las combinaciones de cúbits para determinar el orden en el que estas columnas deben estar en la matriz final. Como se ha comentado, la puerta CNOT trabaja sobre 2 cúbits, por lo

tanto, hay que realizar XOR sobre las posibles combinaciones de 2 cúbits. Entonces debemos realizar XOR con el primer cúbit y el segundo cúbit y el resultado mantenerlo en el segundo cúbit. En la ecuación 2.13 se refleja la operación que se debe realizar y el resultado que se debe esperar.

$$U_{xor}|q_0, q_1\rangle = |q_0, q_r = q_0 \oplus q_1\rangle = |q_0 q_r\rangle, \tag{2.13}$$

siendo $q_0 = |0\rangle\ o\ |1\rangle,\ q_1 = |0\rangle\ o\ |1\rangle\ y\ q_r = |0\rangle\ o\ |1\rangle$.

Por ejemplo, si tenemos dos cúbits con estado $|0\rangle$, la operación será la siguiente: $U_{xor}|0,0\rangle = |0, 0 \oplus 0\rangle = |00\rangle$

En la ecuación 2.14 se pueden observar todas las combinaciones de los 2 cúbits y sus resultados, los cuales, serán el orden en la matriz.

$$\begin{aligned} U_{xor}|0,0\rangle &= |0, 0 \oplus 0\rangle = |00\rangle \\ U_{xor}|0,1\rangle &= |0, 0 \oplus 1\rangle = |01\rangle \\ U_{xor}|1,0\rangle &= |1, 0 \oplus 1\rangle = |11\rangle \\ U_{xor}|1,1\rangle &= |1, 1 \oplus 1\rangle = |10\rangle \end{aligned} \tag{2.14}$$

Por lo tanto, el orden por puestos en las columnas de la matriz CNOT será $|00\rangle$, $|01\rangle$, $|11\rangle$ y $|10\rangle$, tal y como se muestra en la ecuación 2.15. Se debe recordar, con la ecuación 2.12 que cada combinación de 2 cúbits tiene una representación matricial, que es la que luego queda reflejada en cada columna de la matriz 2.15.

$$CNOT = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{2.15}$$

El resultado es la matriz CNOT ya mostrada en la ecuación 2.10

2.3.7 Puerta de Toffoli

La puerta *Toffoli* o la puerta CCNOT trabaja con 3 cúbits, de los cuales los dos primeros son de control. De esta forma, si estos dos cúbits tienen el estado en $|1\rangle$ se realiza el cambio de estado en el último cúbit. Solo si los dos primeros están con estado $|1\rangle$ se realizaría el cambio en el último cúbit, si alguno de los dos primeros cúbits tiene estado $|0\rangle$ ya no se produciría el cambio de estado en el último cúbit. Los dos primeros cúbits únicamente son de control,

afectan al funcionamiento de la puerta, es decir, determinan si se va a producir el cambio o no en el último cúbit.

El circuito de la puerta CCNOT se representa por la figura 2.17 y por la ecuación 2.16.

$$Toffoli(q_{c0}, q_{c1}, q_0) = |q_{c0}\rangle \otimes |q_{c1}\rangle \otimes |q_0\rangle = |q_{c0}q_{c1}q_0\rangle \cdot \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \quad (2.16)$$

siendo $q_{c0} = |0\rangle \ o \ |1\rangle,\ q_{c1} = |0\rangle \ o \ |1\rangle \ y \ q_1 = |0\rangle \ o \ |1\rangle$.

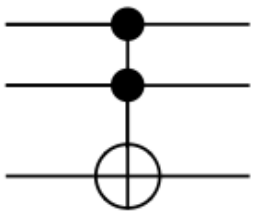


Figura 2.17: Puerta cuántica Toffoli. Obtenida de [48].

Muchas de estas puertas cuánticas pueden verse con más detalle mediante la ejecución de circuitos cuánticos, que se verán en la sección 8.4.

2.4 Problemas de la computación clásica

Los problemas que tienen actualmente los ordenadores clásicos están relacionados con la demanda de las empresas a que estos sean mas veloces. Esta demanda surge debido a que estas empresas desean tratar con cada vez más cantidad de datos y ejecutar algoritmos de complejidad muy elevada.

Para responder a esta demanda, los ordenadores clásicos deben mejorar aspectos necesarios para aumentar la velocidad de procesamiento, para ello existe la **La Ley de *Moore***.

Gordon Moore anticipó que el tamaño de los transistores[2] disminuiría de forma exponencial, permitiendo así que mas transistores puedan ser integrados en los circuitos. De forma que

[2]El transistor es la unidad mas simple de procesamiento de datos. Son como interruptores que bloquean o permiten el paso de los electrones.

mientras más pequeños sean los transistores mayor cantidad de estos se pueden poner en los procesadores y por tanto, se obtendrían procesadores cada vez más rápidos.

En definitiva, la Ley de *Moore* establece que la cantidad de transistores que se podrían añadir en un procesador se duplicaría cada dos años, como se puede observar en la figura 2.18. Esto quiere decir que cada dos años los procesadores aumentarían su velocidad de procesamiento con respecto a los anteriores.

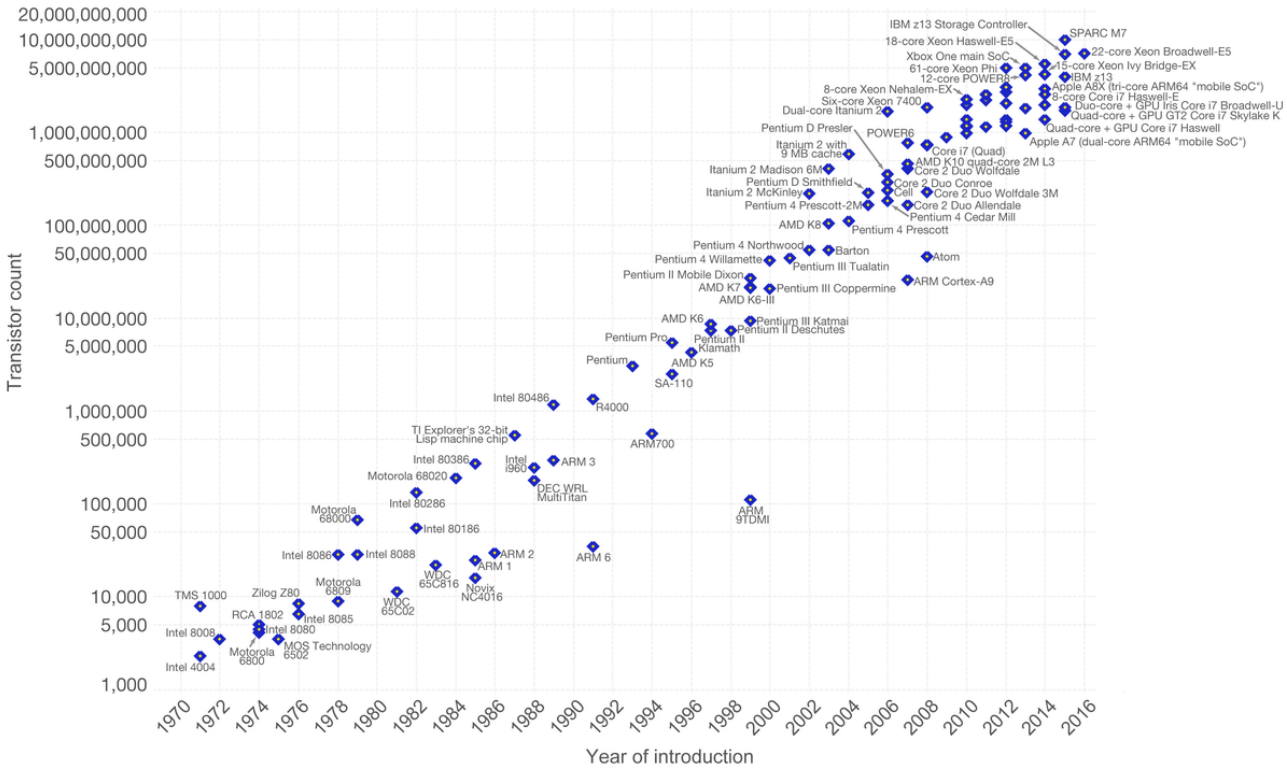


Figura 2.18: Ley de Moore. Obtenida de [59].

En este punto, se puede observar que la velocidad de procesamiento depende directamente de la cantidad de transistores que contamos en un procesador. La solución es sencilla, continuar fabricándolos cada vez con menor tamaño. Sin embargo, los transistores que actualmente se están fabricando son tan pequeños que están rozando el punto de ser intratables.

Actualmente, los transistores se cuentan en el orden de nanómetros, mas concretamente, en el orden de dos nanómetros. En estas escalas, las propiedades de la física funcionan de forma diferente a la que estamos acostumbrados a utilizar, por lo que, crear transistores a estas escalas hace inservible su funcionamiento y por ende, hace inservibles los ordenadores actuales.

Se esta llegando al límite de lo que puede llegar un transistor y como consecuencia, también se esta llegando al límite en el aumento de velocidad de procesamiento actual.

Es a partir de este momento donde no obtendremos mas beneficios con la física actual, es decir, es en este momento en el que el tamaño es tan pequeño, que las leyes por las que se rigen son por las leyes cuánticas.

Con las leyes cuánticas y sus propiedades sobre la materia se puede obtener ese aumento de velocidad de procesamiento que ya no podemos conseguir debido a los problemas actuales con los transistores. Es aquí donde entran en juego todas las propiedades de la computación cuántica que se han explicado anteriormente.

2.5 Problemas de la computación cuántica actual

El principal problema que encontramos en la computación cuántica para los informáticos es el que cita Jose Miguel Piquer, Ph.D. en Informática de la Ecole Polytechnique de París, director Científico y Estratégico del proyecto CIRIC e investigador de la Universidad de Chile:

"*Para los informáticos como yo, la computación cuántica es una espantosa amenaza: el día que exista, echará por tierra toda la computación como la conocemos y habrá que inventar todo de nuevo, desde los lenguajes de programación hasta los sistemas de seguridad y comunicaciones*"[24].

Analizando este problema de cerca, se puede observar que este aspecto conlleva muchos cambios y muchas adaptaciones con respecto a nuestra tecnología actual. El hecho de que la computación cuántica trabaje con cúbits y la clásica con bits, ya es un problema que se tendrá que manejar.

Otro de los grandes problemas de la computación cuántica actualmente es la **decoherencia cuántica**. Esta se da cuando tenemos un estado cuántico entrelazado y pasa a un estado no entrelazado, es decir, pasa a estar en un estado donde se pierden todas las propiedades cuánticas [83]. El cúbit que ya no está entrelazado pierde la propiedad de superposición y como consecuencia, toma un valor único como lo haría un bit clásico. Esto no quiere decir que sea equivalente a un bit clásico, solo que ha perdido las propiedades cuánticas en ese instante de tiempo y el cúbit debe adoptar un valor definido. Además, esto tampoco quiere decir que este cúbit no vaya a poder obtener de nuevo estas propiedades cuánticas. Solo las ha perdido en ese instante debido a la decoherencia cuántica.

Esta decoherencia cuántica se puede dar debido a las perturbaciones que pueden existir en el medio como la intrusión de ondas o fotones. Por lo tanto, un ordenador cuántico debe permanecer aislado de cualquier interferencia externa que pueda perjudicar a este durante la fase de cálculo. Esto significa que se deberá tener unas instalaciones adecuadas y controladas para intentar asegurar el correcto funcionamiento de los cúbits.

Debido a errores en defectos en puertas lógicas cuánticas, en la preparación del estado cuántico inicial, en la medición del cúbit e incluso errores producidos por la decoherencia cuántica, entre otros, existe otro problema a considerar actualmente: la **corrección de errores**. La

corrección de errores es esencial para llegar a una computación cuántica robusta. Actualmente es uno de los mayores desafíos en los que se está trabajando ya que es completamente necesario para aumentar la eficiencia y la aplicabilidad de los cúbits.

Por otra parte, el hardware necesario para establecer un ordenador cuántico es realmente complejo y aún no está definido como tal. Se deben dar unas condiciones ideales para que estos puedan funcionar correctamente:

- Sistema escalable, es decir, si aumenta el problema, se necesitará una mayor cantidad de cúbits para resolverlo y por tanto el sistema cuántico deberá soportar ese aumento de cúbits.
- Se debe seguir una coherencia cuántica, es decir, el sistema debe estar aislado para evitar las interferencias externas comentadas.
- Temperaturas muy bajas y controlables, por ejemplo, los ordenadores cuánticos de *International Business Machines* (IBM) deben funcionar a -273ºC.

Es decir, que las condiciones para poder poner en marcha correctamente un ordenador cuántico son demasiado exigentes actualmente y por lo tanto, con limitadas opciones en su uso.

2.6 Infraestructura de un ordenador cuántico

La arquitectura de un ordenador cuántico no tiene nada que ver con la de uno clásico, el ordenador cuántico no tiene ni memoria RAM, ni disco duro, ni procesador, únicamente posee elementos de dimensiones muy pequeñas.

El computador cuántico posee un **refrigerador de dilución** que emplea helio-3 y helio-4 para alcanzar temperaturas muy bajas(-273ºC) en las cuales se manifiestan las características cuánticas. Dentro del refrigerador se encuentra el **chip superconductor** donde podemos localizar los cúbits. El refrigerador está conectado a un rack donde se sitúan los **generadores y detectores de microondas**, los cuales se encargan de comunicar al ordenador cuántico las operaciones que debe realizar y establecer el estado en el que se encuentra.

Como se ha comentado en la sección 2.5, el ordenador cuántico debe estar en unas condiciones especiales para funcionar. Una temperatura extremadamente baja (-273ºC facilitada por el refrigerador), el aislamiento por las perturbaciones externas que pueden afectar a las propiedades de los cúbits, sin apenas presión atmosférica y aislados del campo magnético terrestre.

Por ejemplo, la empresa IBM, cuenta con estas instalaciones y posee un ordenador cuántico apto para ser usado, como se puede observar en la figura 2.19.

Figura 2.19: Ordenador cuántico de la empresa IBM. Obtenida de [43].

2.7 Usos del computador cuántico

Actualmente los computadores cuánticos están en proceso de investigación, lo que quiere decir que no tienen (de momento) un uso aplicable en nuestra sociedad, pero se puede estimar que en el futuro estos tengan un gran beneficio en algunos ámbitos.

El ordenador cuántico puede ser de gran utilidad para la administración de cuentas de grandes empresas, para cálculos complejos en el campo de la ciencia y la innovación, que se verían particularmente beneficiados acelerando los avances en el límite del conocimiento de muchos campos significativamente más rápido gracias a la velocidad de cómputo del ordenador cuántico. Además, también se verán beneficiadas las disciplinas de la informática, como la ciberseguridad o como la inteligencia artificial, entre otras, que se verán en los capítulos 3 y 8 respectivamente.

Algoritmos cuánticos como "el temple cuántico" nos permiten encontrar valores mínimos de funciones, algo que tiene aplicación directa con la inteligencia artificial y el aprendizaje automático. Al ser tan útil en tareas de aprendizaje automático, sus aplicaciones en medicina, química, criptografía, estudio de moléculas complejas, etc. pueden obtener un gran beneficio utilizando esta tecnología [52].

La industria farmacéutica también se vería beneficiada ya que, por ejemplo, para la investigación de medicamentos hay que considerar diferentes combinaciones de variables y un ordenador cuántico podría realizar multitud de combinaciones en un tiempo más que aceptable.

A día de hoy, el ordenador clásico no está pensado para ser sustituido por el ordenador cuántico en un ámbito personal ya que las condiciones de un usuario corriente no son las adecuadas para poner en marcha un ordenador cuántico. En cambio, sí está pensado para todos esos entornos que requieren de gran capacidad de cómputo donde el tiempo de espera en la ejecución de los algoritmos son clave. En definitiva, todo trabajo que requiere gran cantidad de cálculo y un tiempo alto de espera de ejecución del algoritmo se podrá beneficiar de la computación cuántica.

A continuación se va a presentar un proyecto que se llevaba mucho tiempo realizando y que con la computación cuántica, quizás, el tiempo requerido hubiera sido mucho menor para completarlo.

2.7.1 Proyecto del genoma humano

El 31 de marzo de 2022 se completó el **proyecto del genoma humano** que comenzó en el año 1990. Han pasado 32 años y se han requerido una cantidad de datos enormes para completarlo.

En España, las máquinas encargadas de realizar el proceso de tratamiento de los datos para este proyecto son las del Centro Nacional de Análisis Genómico (CNAG). Estas trabajan para secuenciar 20 genomas al día y cada genoma cuenta con 3.300 millones de moléculas químicas. Para obtener buenos resultados, se secuencian 2 billones de moléculas químicas al día. Esto solo es el paso de secuenciación, después se deben realizar otros pasos donde se pueden encontrar mutaciones que podrían significar cáncer u otras enfermedades. Todo este proceso conlleva un análisis de 30 terabytes de datos, por lo que ordenadores normales no podrían resolver esta problemática [45].

Para poder realizar este trabajo de análisis se requiere de un superordenador capaz de manejar con tal cantidad de datos.

Al final, el objetivo de este proyecto es poder determinar que gen es el que ha causado la enfermedad para producir un tratamiento mas exacto. Y posteriormente, investigar que medicamentos pueden tratar cada mutación.

Se han requerido de 32 años de análisis de datos en ordenadores especializados para poder completar el proyecto. Sabiendo que la computación cuántica puede consumir mucho menos tiempo manejando la misma cantidad de datos que la computación clásica, es muy probable que habiendo tenido la computación cuántica a nuestro alcance se podría haber agilizado en gran medida este proceso de análisis de datos.

3 Computación cuántica en la seguridad informática

La seguridad informática es un área de la informática que trata la protección de la información y el procesamiento que se hace con esta con la finalidad de proteger tanto a las personas como las infraestructuras y sus datos de amenazas externas.

La seguridad informática abarca diferentes tipos:

- **Seguridad de hardware**: protegiendo dispositivos físicos mediante proxy, firewall, etc.
- **Seguridad de software**: protegiendo los sistemas contra ataques maliciosos relacionados con las vulnerabilidades de los programas.
- **Seguridad de red**: protegiendo los datos que pueden ser accesibles a través de la red.

En la seguridad informática se pueden encontrar diversos métodos para conceder seguridad a los diferentes niveles que se han comentado anteriormente, pero se va a estudiar con más profundidad la ciencia de la criptografía ya que será la que más cambios tendrá con la llegada de la computación cuántica. Esto es debido a que la criptografía esta ampliamente utilizada mediante algoritmos matemáticos. La robustez de estos algoritmos criptográficos dependen directamente de su diseño y de la longitud de la clave a cifrar. Averiguar la clave original mediante una clave cifrada con métodos como el **Ataque por fuerza bruta** supondrían un coste computacional enorme y por tanto un tiempo de espera muy alto para obtener la clave original. Cuanto mejor sea el diseño del algoritmo y mayor sea la longitud de la calve a cifrar, mas tiempo se requerirá para poder averiguar la clave cifrada. Este tiempo sería mucho menor debido a que la computación cuántica podría resolver estos algoritmos matemáticos mucho mas rápido. A lo largo de esta sección se podrá ver como algoritmos actualmente muy robustos, como RSA, pueden ser vulnerables con la computación cuántica.

3.1 Criptografía clásica

La criptografía tradicional es la ciencia que se encarga de transformar el contenido de un mensaje para que cuando se transmita por un canal hasta su destinatario solo sea capaz de descifrarlo dicho destinatario.

En la criptografía podemos encontrar varios elementos imprescindibles: el mensaje que se ha de transmitir, el cifrado (proceso que codifica el mensaje), el descifrado (proceso de obtención del mensaje a partir del mensaje cifrado) y la clave (datos necesarios para cifrar o descifrar).

Actualmente, la criptografía es una rama de las Matemáticas que utiliza métodos y cálculos matemáticos para obtener el cifrado, el descifrado y las claves.

En cuanto a los tipos de criptografía podemos encontrar:

- **La criptografía simétrica**: algoritmos que usan una misma clave para cifrar y descifrar. (DES, AES, Blowfish, IDEA, etc.).
- **La criptografía asimétrica**: tanto receptor como destinatario tienen dos pares de claves, una pública y una privada. El receptor tiene su clave privada y la clave pública del destinatario y el destinatario tiene su clave privada y la clave pública del receptor. La clave pública se transmite de la misma forma que el mensaje, pero la clave privada se mantiene oculta. (Diffie-Hellman, RSA, etc.).
- **Hash**: algoritmo que convierte un bloque de datos en una nueva serie de caracteres de longitud fija, independientemente del bloque de datos entrada. (MD5, SHA-1, etc.).

Estos tipos de criptografía están pensados para que el tiempo necesario para descifrar un mensaje (sin las claves de cifrado) sea considerablemente alto, ya que se basan en combinaciones matemáticas complejas donde se requiere una gran cantidad de cálculos inviables de realizar en un tiempo razonable.

En el caso del algoritmo RSA, por ejemplo, para averiguar la clave de cifrado hay que realizar una factorización en números primos, lo cual, si el número es realmente grande, el cómputo puede llevar mucho tiempo, en el orden de años.

Es aquí donde radica el problema de los esquemas criptográficos, que se basan en problemas matemáticas donde la seguridad se contempla en el hecho de que no se pueden resolver estos problemas, es decir, tienen solución y se puede llegar hasta ella, pero la complejidad es tan alta para los ordenadores actuales que es prácticamente imposible solucionarlos.

La llegada de la computación cuántica pone en jaque a estos sistemas criptográficos basados en el tiempo de cómputo necesario para obtener las claves puesto que, como hemos hablado anteriormente, los ordenadores cuánticos pueden realizar muchas más operaciones en menor tiempo que los ordenadores actuales.

Una vez estuviera asentada la computación cuántica, estos sistemas dejarían de ser seguros porque, como ya se ha visto, la velocidad de cómputo de un ordenador cuántico es exponencialmente mejor a la de un ordenador clásico. Lo que por *fuerza bruta* con la computación clásica se resolvía en un tiempo desmesurado, con la computación cuántica se tardarían horas en obtener el resultado [53].

Muchos de los esquemas de cifrado de clave pública, que hoy en día son los más usados, como *RSA*, *Diffie-Hellman* y la curva elíptica, dejarían de ser útiles porque no podrían asegurar la seguridad de la comunicación.

Por lo tanto, el cifrado debe ser seguro contra las computadoras cuánticas. Es decir, la criptografía clásica no serviría con la llegada de esta nueva tecnología y por lo tanto, se debe desarrollar técnicas de criptografía avanzadas compatibles con la computación cuántica. Será necesario realizar una nueva criptografía y realizar nuevos algoritmos aptos para la computación cuántica. Por ello, surgió la idea de la criptografía cuántica, que trataremos a continuación.

3.2 Criptografía cuántica

La idea de la **criptografía cuántica** nació en 1970. Esta criptografía utiliza los principios de la mecánica cuántica para garantizar absoluta seguridad en la información transmitida.

Se ha de recordar que la mecánica cuántica es una parte de la física que estudia la materia a nivel molecular, atómico y subatómico y nos permite estudiar el comportamiento de esta materia con dimensiones muy pequeñas.

La criptografía cuántica tiene el mismo objetivo que la criptografía clásica, pero esta emplea los principios de la mecánica cuántica para cifrar los mensajes, de forma que solo puede descifrarlos de manera correcta un destinatario determinado. Si un atacante viese el mensaje, al "observarlo", la información se alteraría debido a las propiedades de los cúbits, por lo tanto, la información que recibiría el atacante sería distinta a la que se envió desde un principio.

La primera distinción que encontramos entre criptografía clásica y la cuántica es que la clásica utiliza operaciones matemáticas y en la criptografía cuántica se emplean los fotones y sus propiedades cuánticas para cifrar los mensajes.

¿Cómo funciona entonces la criptografía cuántica?

En la criptografía cuántica se utiliza la técnica *Quantum Key Distribution* (QKD) para crear claves de cifrado empleando fotones y sus propiedades cuánticas. Estas propiedades cuánticas se refieren a la "superposición cuántica" y el "entrelazamiento cuántico" que ya se habían comentado en la sección 2.2.1 y 2.2.2, respectivamente.

A continuación se va a realizar una explicación del funcionamiento de QKD mediante un ejemplo [38]:

Imaginemos que tenemos dos interlocutores, Alice y Bob. Alice, en este caso, es la que emite los mensajes. Inicialmente, Alice envía una serie de cúbits a Bob. Bob, cuenta con unos detectores capaces de medir estos cúbits. Una vez se realiza dicha medición[1], Bob le cuenta

[1]Se ha de recordar que la medición es una propiedad cuántica vista en la sección 2.2.1

a Alice el resultado de la medición de los cúbits y los compara con sus mediciones. En este momento, Alice le cuenta donde ha fallado en las mediciones de sus cúbits, de esta forma Bob es consciente de donde se ha realizado mal la medición. Las mediciones de cúbits de Bob y Alice que no concuerdan se desechan. De forma que de la serie de cúbits enviados inicialmente solo quedará un conjunto de cúbits para Alice y Bob, siendo este conjunto el mismo para los dos. Es este conjunto de cúbits el cual servirá como clave para Alice y Bob para poder realizar la encriptación y desencriptación de los mensajes. Actualmente, estos mensajes se siguen transmitiendo de forma tradicional, es decir, con bits, pero con el detalle de que la clave ya no es tan sencilla de interceptar.

Si un intruso intenta "observar" (propiedad cuántica comentada en la sección 2.2.1, nombrada técnicamente como "medición") la etapa de "negociación de la clave", el estado de los cúbits se vería alterado y por lo tanto, la secuencia que Alice mandó en un principio no sería la misma que Bob recibiría. De esta forma si Bob envía un trozo de la secuencia, que ha recibido alterada, a Alice, ella podría comprobar que hay un intruso en la comunicación ya que la secuencia que Alice envió no coincide con la secuencia que Bob aún sin realizar la medición le ha vuelto a enviar.

Este proceso de elección de la clave entre interlocutores se realiza cada pocos segundos, por lo tanto, las claves de encriptación se renuevan evitando así posibilidades de intrusos en las comunicaciones. Debido a estos beneficios, la criptografía cuántica podría suponer una solución a los intentos de los ciberdelincuentes para acceder a la información cuando viaja de emisor a receptor.

La figura 3.1 muestra el proceso que se realiza y se observa como el intruso no puede obtener la clave cuántica y por tanto no puede descifrar el mensaje.

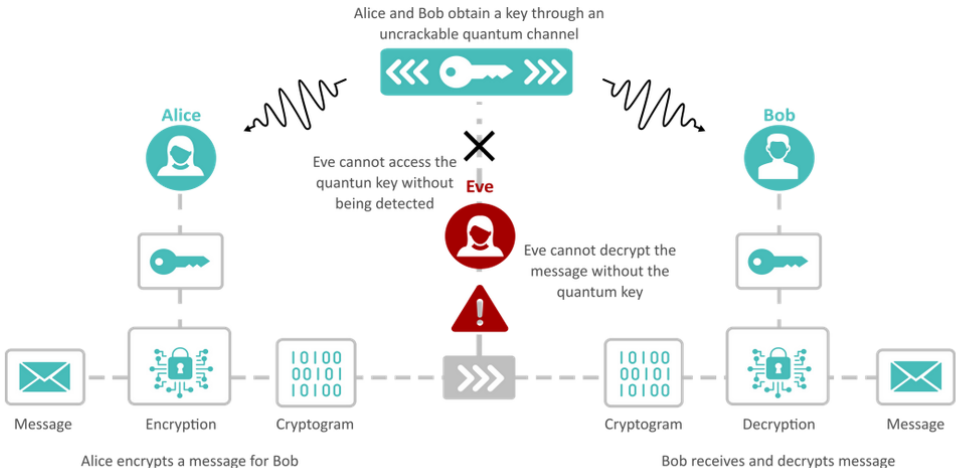


Figura 3.1: Proceso de *Quantum Key Distribution*. Obtenida de [67].

Los beneficios de la criptografía cuántica son muy satisfactorios, pero, por otra parte, la computación cuántica conlleva problemas importantes para la criptografía clásica actual

basada en las matemáticas. Por ello surge el concepto de **criptografía post-cuántica**, que estudiaremos a continuación.

Como prueba del riesgo que supone la computación cuántica contamos con el **algoritmo de Shor** (algoritmo que se explicará en la sección 6.2.2) el cual, a grandes rasgos, se encarga de descomponer en factores un número N con una complejidad temporal de $O((logN)^3)$ y espacial de $O(logN)$ pudiéndose ejecutar sobre circuitería cuántica. Debido a este algoritmo y al potencial de la computación cuántica, la criptografía de clave pública ampliamente utilizada, como en el caso de RSA, llegaría a estar obsoleta.

3.3 Criptografía post-cuántica

La criptografía post-cuántica surge debido a la necesidad de crear algoritmos capaces de soportar todo el potencial de un ordenador cuántico incluyendo también soluciones basadas en mecánica cuántica y cifrado cuántico para contrarrestar estos posibles ataques con la tecnología actual.

El tiempo que duraría la criptografía post-cuántica estaría entre el periodo de instauración de los ordenadores cuánticos en la tecnología actual y la creación de algoritmos nuevos y resistentes capaces de resistir un ataque de un ordenador cuántico sobre la criptografía clásica.

Actualmente existen diferentes estrategias para poder defenderse de la llegada de los ordenadores cuánticos pero, en general, estas estrategias no están preparadas para lo que estaría por venir. Se requiere de más tiempo para perfeccionar y encontrar nuevas técnicas eficientes y seguras en el tiempo que dure la criptografía post-cuántica.

A continuación se presentan algunas de estas estrategias [96]:

3.3.1 Criptografía basada en retículos

Los retículos son un conjunto de puntos que se presentan de forma regular en un plano n-dimensional infinito. Al ser este plano infinito y teniendo el ordenador una capacidad de recursos finitos se necesita que dichos retículos se representen mediante “bases”.

Las bases son un conjunto de vectores utilizados para representar cualquier punto del plano que forma dicho retículo. A partir de las bases se puede expresar cualquier vector en un espacio vectorial como combinación lineal de los elementos de esta base.

Para representar un punto del plano puede existir mas de una base posible. Cuanto más pequeño (longitud de la representación en un plano) sean los vectores de las bases mas fácil será determinar la estructura del retículo.

Es aquí donde reside la importancia en esta estrategia, ya que si contamos con vectores

suficientemente grandes en longitud, más difíciles serán de determinar la estructura de los retículos.

3.3.2 Criptografía basada en funciones polinomiales multivariables

Esta estrategia utiliza criptografía de clave asimétrica y está basada en polinomios de diferentes variables establecidas en un campo finito. Estos sistemas criptográficos se basan en la resolución de problemas de estos polinomios. El termino multivariable se refiere a que los polinomios que se utilizan tienen mas de una variable, por ejemplo, como la representada en la ecuación 3.1.

$$f(x, y) = x \cdot y^2 + x \cdot y + y \tag{3.1}$$

En general, un esquema multivariable se construye a partir de una función relativamente fácil de resolver que es "camuflada" entre dos transformaciones lineales para así obtener un nuevo sistema. El nuevo sistema representaría la clave pública y el sistema original junto con las transformaciones representarían la clave privada.

3.3.3 Criptografía basada en funciones hash

Esta criptografía funciona muy parecido a una firma digital pero para esta estrategia se hace uso del **árbol de *Merkel* o árbol hash**. El árbol de *Merkel* es una estructura de datos en árbol en el que cada nodo no hoja tiene etiquetado el hash de la concatenación de las etiquetas o valores de sus nodos hijo.

Esta estructura permite que se puedan agrupar los hashes de distintos bloques de datos en un solo hash (en el nodo principal del árbol) con el inconveniente de aumentar el tamaño de la salida. Si el "hash padre" es quien contiene el resto de hashes de sus hijos, el tamaño de la salida crecerá según la cantidad de hijos no hoja que posea el árbol.

En la figura 3.2 se puede observar como sería un árbol de *Merkel*.

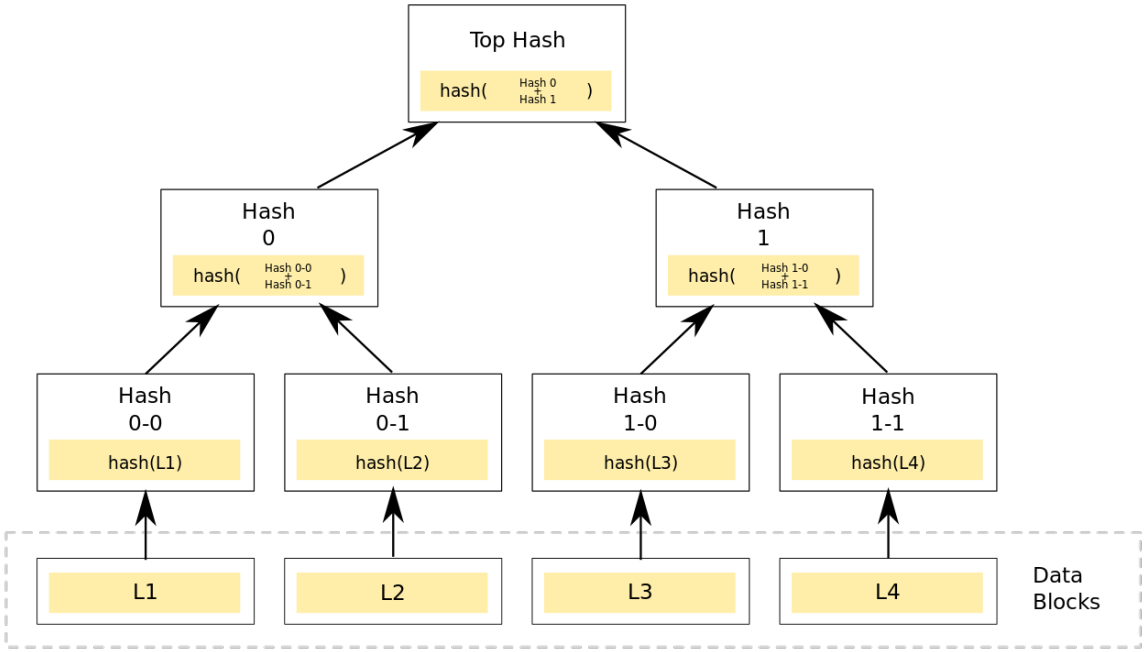


Figura 3.2: Árbol de *Merkel*. Obtenida de [88].

3.3.4 Criptografía basada en códigos con corrección de errores

Esta estrategia fue recomendada por el *Post Quantum Cryptography Study Group* como un aliado para evitar los ataques cuánticos.

En las comunicaciones suelen producirse errores haciendo que el mensaje que se envía a un destinatario sea diferente al enviado. Estos errores suelen deberse a interferencias externas que alteran este mensaje. Para mejorar la calidad en las comunicaciones contamos con códigos capaces de realizar corrección de errores. Estos pueden arreglar de forma automática un mensaje erróneo gracias a la redundancia de la información.

Estos códigos que analizan y añaden redundancia en los mensajes deben ser conocidos por los dos miembros de la comunicación. Los códigos mas comúnmente utilizados en la criptografía son *Walsh-Hadamard*, *Hamming* y *Goppa*, entre otros.

La idea de la criptografía basada en códigos con corrección de errores es producir estos errores (no necesariamente tienen que ser provocados por factores externos), es decir, de añadir errores a conciencia en los mensajes cifrados. De esta forma, el receptor es el único con la capacidad para arreglar este mensaje y obtener la información.

3.4 Computación cuántica en las criptomonedas

Las criptomonedas son monedas digitales las cuales mediante técnicas de criptografía proporcionan en el comercio un sistema de pago seguro, tanto para cobros como para pagos.

Todas estas transacciones de cobros y pagos de criptomonedas se registran en una especie de "libro mayor o libro contable" digital descentralizado llamado ***blockchain***, de forma que se pueden observar los registros de las criptomonedas. Cabe destacar que cada criptomoneda tiene su propio *blockchain*.

Con la llegada de la computación cuántica, las criptomonedas tienen dos inconvenientes a tener en cuenta y por lo que podría ser la decadencia de estas monedas digitales:

- Una forma de obtener las criptomonedas es realizando el "minado" de estas, que al fin y al cabo es resolver complicados problemas matemáticos y obtener dicha moneda digital como recompensa. Este "minado" puede conllevar mucho tiempo de computación para una recompensa muy pequeña. De forma que muchas soluciones para agilizar este proceso han sido la de comunicar y "trabajar en equipo" con diferentes nodos para aumentar la velocidad a la que se resuelven los problemas y por lo tanto aumentar la velocidad de producción de estas monedas.

  La computación cuántica marca un antes y un después ya que este cómputo necesario es totalmente viable y alcanzable con los ordenadores cuánticos sin la necesidad de tener un "trabajo en equipo" con diferentes nodos. Esto puede suponer un gran riesgo ya que

muchas organizaciones con los dispositivos adecuados pueden hacerse con el control de la ganancia de las criptomonedas y ser las que "controlen" el *blockchain*.

Esto se conoce como el **Ataque del 51%**, es decir, si una organización controla más del 50% de la red de minado, es como si controlara de manera unilateral toda la *blockchain*. Esto supondría que esta organización controlaría las validaciones de transacciones. Para dar por válido o cierta alguna transacción en la *blockchain*, la mayoría de los nodos de esta deben "estar de acuerdo" (**protocolos de consenso**) [35].

Al tener la organización la mayoría de los "votos", esta sería quien tendría el control sobre la red y las transacciones que se producen en esta.

Esto es un grave problema ya que con un ordenador cuántico es perfectamente posible obtener dicho 51% gracias a que se cuenta con gran capacidad de procesamiento y por lo tanto más cantidad de posibles nodos activos [97].

- La seguridad de las criptomonedas se basa en la criptografía tan potente con la que actualmente cuenta, pero esta criptografía, ya mencionada anteriormente, esta basada en las matemáticas y descifrarla es prácticamente imposible con un ordenador convencional, pero no con un ordenador cuántico. Por lo tanto, estas transacciones serían vulnerables y podrían estar expuestas a ataques por parte de la tecnología basada en computación cuántica.

Esos inconvenientes son una gran amenaza para el mundo de las criptomonedas pudiendo dejarlas inhabilitadas por su escasa seguridad. Algunos expertos[2] exponen que con la llegada de la computación cuántica, las criptomonedas deberán adaptarse a los cambios criptográficos o sufrirán las consecuencias de esta [97].

Actualmente, con un ordenador cuántico de 100 o 300 cúbits es una tarea difícil romper la seguridad de las criptomonedas, pero con el paso del tiempo y la evolución de esta tecnología podría ser perfectamente plausible. Por lo tanto, es totalmente necesario desarrollar medidas criptográficas para prevenir esta situación.

3.5 Internet de las cosas cuántico

El internet de las cosas, *Internet of Things* (IoT) es la agrupación e interconexión de dispositivos y objetos a través de una red, donde todos estos podrían ser visibles e interaccionar entre todos. Cualquier cosa que se pueda conectar a internet e interaccionar sin necesidad de la intervención humana será considerada como nodo de interacción de máquina a máquina,"M2M".

Actualmente existen diferentes problemas en la seguridad de los sistemas IoT como la

[2] *Divesh Aggarwal*, investigador de la Universidad Nacional de Singapur o *Jian-Wei Pan*, creador de un ordenador cuántico de 66 cúbits.

violación de los datos, la autenticación, ataques de canal lateral, actualizaciones irregulares, *malware* y *ransomware*.

Con la computación cuántica, podemos abordar los problemas que obstaculizan el crecimiento del IoT. Gracias a la computación cuántica podemos conseguir:

- Potencia de cálculo compleja optimizada: la velocidad conseguida es increíblemente alta. IoT se beneficia de la alta velocidad ya que estos generan gran cantidad de datos que deben ser tratados y enviados a los destinatarios correspondientes. Hay algunos ejemplos de dispositivos IoT donde la velocidad no es prioritaria como sería en el caso de la domótica. Sin embargo, existen muchos otros como sensorización, sistema de alertas o recopilaciones de datos en tiempo real donde el tiempo de envío de los datos que los dispositivos IoT realizan es importante.

  Sin ir más lejos, la Universidad de Alicante cuenta con un proyecto llamado *Smart University* donde se realiza una gran ingesta de datos a través de dispositivos IoT. Estos dispositivos envían los datos en periodos de tiempo muy pequeños para poder obtener unos resultados en tiempo real. El aumento de la velocidad de estos dispositivos IoT debido a la computación cuántica podría mejorar la exactitud de dichos resultados.

- Proceso de validación y verificación más rápido: se garantiza una optimización constante de los sistemas gracias a la aceleración en los procesos de verificación y validación en todos los sistemas.

- Comunicaciones seguras: por medio de la criptografía cuántica aumentamos la seguridad de la comunicación para evitar ciberataques, violaciones de datos, autenticación, *malware* y *ransomware*.

Cabe recordar que la computación cuántica puede dañar la seguridad criptográfica con la que contamos con la tecnología actual. Las comunicaciones también dejarían de ser seguras para todos los dispositivos IoT que dependan de la criptografía clásica. Este aspecto es bastante importante en el campo IoT ya que el principal trabajo de los dispositivos IoT es el de comunicarse, por lo tanto, proyectos como el de *Smart University* donde se establecen comunicaciones constantemente podrían contar con una brecha de seguridad a tener en cuenta.

4 Computación cuántica en redes informáticas

En la actualidad, una red informática es un conjunto de dispositivos interconectados entre sí a través de un medio tanto cableado como inalámbrico, donde se puede intercambiar información y compartir recursos entre los diferentes dispositivos de la red. En las redes informáticas intervienen diferentes elementos como los servidores, los clientes, los medios de transmisión y los elementos hardware y software.

Algunos tipos de redes informáticas actualmente existentes son PAN, WPAN, LAN, WLAN, CAN, MAN, WAN y VLAN.

- Redes PAN: Las redes PAN o redes de área personal son redes para uso personal, están limitadas a un espacio en concreto y la conexión entre los dispositivos se realiza de forma cableada.

- Redes WPAN: Las redes WPAN o redes de área personal inalámbricas son como las redes PAN pero la conexión no se realiza vía cable sino por medio de ondas electromagnéticas. *Bluetooth* sería la red WPAN más utilizada.

- Redes LAN: Las redes LAN o redes de área local son muy parecidas a las redes PAN pero cuentan con protocolos más estandarizados y trabajados para su uso corriente. Estas pueden llegar a velocidades mucho más altas que las redes PAN

- Redes WLAN: Las redes WLAN o redes de área local inalámbricas son como las redes LAN pero la conexión no se realiza vía cable sino por medio de ondas electromagnéticas

- Redes CAN: Una red CAN o red de área de campus es una red que conecta diferentes redes de área local que están limitadas por un espacio geográfico determinado, como por ejemplo, una universidad.

- Redes MAN: Las redes MAN o redes de área metropolitana son redes que engloban diversas LANs y/o diversas CANs. El tamaño de estas redes puede dar lugar a conexiones a nivel de ciudad o municipio.

- Redes WAN: Las redes WAN o redes de área amplia son redes que pueden englobar diversas LANs, CANs y/o MANs. Esta tecnología no conecta ordenadores individuales sino que conecta redes con otras redes

- Redes VLAN: Las redes VLAN o redes de área local virtuales permiten que un dispositivo se pueda conectar a una red en concreto sin necesidad de estar conectado a ella

de forma directa. La conexión a esa red se realiza a través de otra red.

El funcionamiento de estas redes informáticas está definido por diversos estándares, pero el más extendido es el TCP/IP que se basa en el modelo OSI.

Gracias a las teorías establecidas en las redes informáticas, entra en juego "internet". Internet es un conjunto de redes interconectadas que utilizan el protocolo TCP/IP que comunican servidores y clientes. En internet podemos encontrar servicios como WWW, SMTP, FTP, P2P, SSH, entre otros, y para cada servicio se establecen diferentes protocolos y estándares que servidor y cliente deben conocer y cumplir para realizar exitosamente la comunicación. La computación cuántica trae consigo una mejora de la red informática actual, el **internet cuántico**.

4.1 Internet cuántico

El internet cuántico es una red que permitiría a dispositivos cuánticos intercambiar información que aprovecha las leyes de la mecánica cuántica, es decir, cumple el mismo objetivo que Internet actual, pero aprovechándose de las propiedades de la mecánica cuántica.

En teoría, el internet cuántico tendrá una capacidad muy superior al internet actual pudiendo llegar a niveles de funcionamiento inalcanzables para la multitud de servicios existentes en la red.

El internet cuántico, en términos generales, permitirá transmitir los cúbits a través de una red de dispositivos cuánticos separados físicamente gracias a las propiedades de los cúbits anteriormente vistas y es gracias a esto por lo que la seguridad informática con el internet cuántico será mucho más prometedora.

La red cuántica por la que viajarían los datos sería muy segura debido a que las comunicaciones podrían contar con la criptografía cuántica mencionada en el capítulo 3. Esto quiere decir que las comunicaciones en dicha red contarían con la seguridad que aporta la criptografía cuántica. Expertos consideran que la computación cuántica resultaría de muchísima ayuda a la hora de crear un sistema de comunicaciones completamente nuevo y seguro.

Estos beneficios en el internet cuántico se deben al uso de los cúbits ya que estos podrían transportar los datos de un punto a otro casi instantáneamente y sin alteraciones en el contenido del mensaje debido a posibles interferencias. Esto podría darse gracias a que los cúbits cuentan con la propiedad de la **teletransportación cuántica** comentada en la sección 2.2.2.1.

Una de las principales ventajas del internet cuántico vendría por la creación, almacenamiento y movimiento de la información con muchísima facilidad y sin las limitaciones típicas de las redes actuales, debido a problemas de interferencias, protocolos, perdida de paquetes, fragmentaciones, lentitud, latencia, entre otros.

Esta red actualmente es solo una teoría, pero algunos expertos afirman que el internet cuántico está actualmente en construcción. Se cree que para 2024-2025 estaría disponible una primera versión de la red cuántica [12].

4.1.1 Experimento de internet cuántico

El 8 de febrero de 2021 se publicó que un equipo de físicos había creado una red cuántica donde intervenían tres dispositivos cuánticos interconectados mediante **entrelazamiento cuántico** [36].

Cada uno de los dispositivos utilizados en el experimento mencionado cuenta con un cúbit que permite dicho entrelazamiento entre los otros dos. Esta red puede ser el inicio de la creación del futuro internet cuántico.

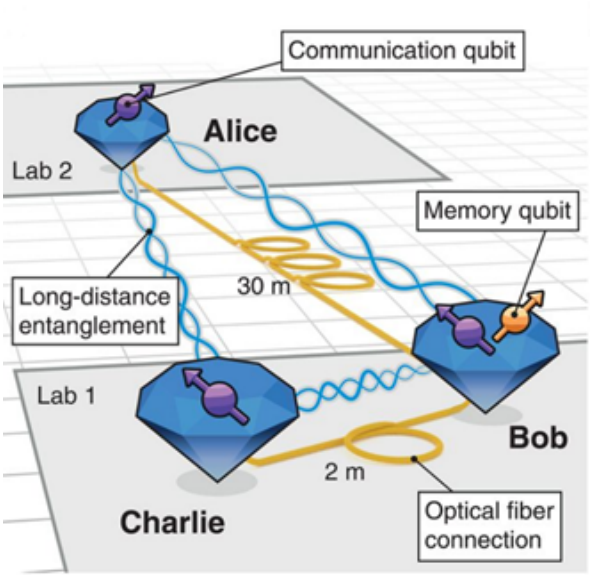


Figura 4.1: Experimento cuántico. Obtenida de [27].

Hasta ahora, el grupo de la Universidad Tecnológica de Delft dirigido por el físico *Ronald Hanson* logró conectar tres dispositivos de forma que los cúbits de dos nodos de la red estuvieran siempre entrelazados. Al mismo tiempo, los cúbits de los tres nodos están también en un estado de entrelazamiento triple, lo que permite a los tres usuarios tener una conexión entre ellos e intercambiar información secreta. Esta información es secreta gracias a que los nodos cuentan con entrelazamiento cuántico, el cual permite aplicar la criptografía cuántica comentada en el capítulo 3. Cabe recordar que el entrelazamiento cuántico es una propiedad de los cúbits ya comentada en la sección 2.2.2.

Cada uno de los nodos almacena la información cuántica en un cristal sintético de diamante. Además, cada uno de los cúbits de cada nodo puede emitir un fotón que se envía a sus vecinos por medio de un cable de fibra óptica produciendo así el entrelazado entre los cúbits.

Bob, en la figura 4.1, también se preparó para poder almacenar información en una **memoria cuántica**. La responsabilidad de este almacenamiento de la información es del cúbit naranja representado en la figura. Este cúbit adicional se obtuvo debido a manipulaciones en los electrones sobre el nodo Bob. Gracias a la memoria cuántica, estos tres dispositivos pudieron conectarse en red. No obstante, este cúbit debe estar lo suficientemente bien aislado de su entorno para que su estado cuántico sobreviva y poder mantener así la red cuántica lo máximo posible.

Estas memorias cuánticas pueden mantener sus estados durante más de un minuto, lo cual puede parecer muy poco tiempo, pero es una eternidad para los estándares del mundo subatómico. De momento, no se ha podido realizar el almacenamiento de información con más tiempo, pero este hecho es el principio de una posible ampliación o incluso de nuevas técnicas de almacenamiento que podrían surgir en un futuro no muy lejano.

Una vez se cuenta con este cúbit especial, el procedimiento para montar la red cuántica es el siguiente:

Inicialmente, el cúbit que no realiza la función de almacenamiento (el cúbit morado en la figura 4.1) de Bob se entrelaza con el cúbit de uno de los otros nodos, por ejemplo, con el de Alice. Seguidamente, el cúbit de memoria cuántica (el cúbit naranja en la figura 4.1) almacena el estado cuántico del cúbit morado de Bob, de forma que Bob sigue en conexión con Alice y ahora tiene disponible el cúbit morado de nuevo para entrelazarse con el cúbit de Charlie.

El importante descubrimiento de este experimento fue la técnica de **transferencia de entrelazamiento** que permite almacenar la información en uno de los nodos, esto podría resultar esencial para una futura **internet cuántica** análogo al funcionamiento de los enrutadores para el internet actual.

Este grupo de investigadores no fue el primero en lograr realizar una red de nodos cuánticos. En 2019 otro equipo de la Universidad de Ciencia y Tecnología de China ya lo consiguió utilizando otro tipo de cúbits basados en nubes de átomos en vez de este con átomos individuales localizados en un sólido, pero este experimento de 2019 no permitía entrelazamiento a voluntad por lo que al detectar fotones los investigadores solo podían deducir a posteriori el entrelazamiento que había. Por otra parte, *Rodney Van Meter*, ingeniero de redes cuánticas de la Universidad de Keio en Tokio, añadía que los cúbits basados en nubes de átomos eran más difíciles de manejar y serían un obstáculo para el entrelazamiento. Los cúbits de nubes de átomos son más limitados en lo que pueden realizar en cuanto al entrelazamiento [36].

El experimento de 2021 tiene impedimentos debido a las limitaciones de los materiales y átomos utilizados, por lo que el equipo nuevamente está intentando investigar en la misma línea con diferentes materiales previsiblemente más eficaces.

Este experimento ha marcado un antes y un después en la computación cuántica y podría ser el inicio que siente las bases del **internet cuántico** en el futuro.

4.1.2 Ventajas del Internet cuántico

El internet cuántico traerá multitud de beneficios, pero la ventaja más notoria es la reducción de la **latencia**.

Teóricamente, al estar los cúbits entrelazados, la comunicación entre estos es instantánea incluso a grandes distancias. Pudiendo llegar a kilómetros de distancia, por ejemplo, con cúbits teletransportados a través de una fibra óptica [42]. Recordando lo que producía el entrelazamiento en la sección 2.2.2, los cúbits, al estar entrelazados, la acción en un cúbit debe tener un efecto inmediato en el otro sin importar la distancia.

Por otra parte, la **seguridad** es otro de los elementos más importantes en el internet cuántico donde es prácticamente imposible espiar una conversación con datos cuánticos, porque como ya se había comentado anteriormente, durante este capítulo y en el capítulo 3, si se observa el estado de un cúbit, este cambia.

El simple hecho de que un intruso este en la red y pueda observar la comunicación alteraría los mensajes de emisor a receptor, pudiendo, evidentemente, comprometer la finalidad de la comunicación pero asegurando la seguridad de esta. Una vez detectado al intruso, para no entorpecer el funcionamiento de la comunicación, el paso siguiente sería expulsarlo de dicha red cuántica ya que si permaneciera en esta, los mensajes nunca llegarían íntegros al destinatario ya que, como se ha comentado, al "observar" la información cuántica, esta sería diferente para el destinatario. Es por ello que uno de los problemas que surgían era la imposibilidad de copiar ni amplificar estos datos.

Por otra parte, la memoria de los cúbits sigue siendo un factor importante para la creación de una red cuántica estable y funcional.

Para conseguir esta red cuántica habrá que continuar investigando, perfeccionando la codificación, el almacenamiento y la transmisión de la información cuántica, incluyendo también la creación de un equivalente a los **protocolos de red** que se utiliza en la actualidad, que por el momento no se ha desarrollado, pero que actualmente se está trabajando en ello.

5 Computación cuántica en lenguajes de programación

Actualmente, un lenguaje de programación es una forma de comunicarse entre el programador y la máquina, indicando a la maquina todas las instrucciones que debe de seguir. El programador debe estructurar la comunicación la cual está conformada por símbolos, palabras claves, reglas semánticas y sintácticas para permitir el entendimiento entre un programador y una máquina.

Existen diversos lenguajes de programación como C, C++, C#, Java, PHP, JavaScript, Python, etc.

En cambio, la computación cuántica, al estar actualmente en desarrollo, existen muy pocos lenguajes comparados con la computación actual. Hay que tener en cuenta que la estructura de programación y la lógica de programación que hasta ahora en la computación actual tenemos, no tendría porque ser del todo similar en la computación cuántica porque si recordamos, en la computación cuántica contamos con los cúbits y no con los bits. Cuando se programa, todo el código se traduce a binario, es decir, todas las ordenes que mandamos al ordenador se traducen en bits para que el procesador trabaje con ellos. Sin embargo, como ya se ha visto, los bits no actúan igual que los cúbits . Como consecuencia, la forma de programar probablemente tomará una corriente diferente a la actual. Debido a que la computación cuántica no esta del todo avanzada es difícil determinar actualmente si la forma de programar cambiará o simplemente el proceso de compilación es el que cambie. Lo que es evidente es que algo deberá cambiar para que pueda programarse de forma sencilla e intuitiva con la computación cuántica.

5.1 Lenguajes cuánticos de programación

Actualmente investigadores han desarrollado varios lenguajes para aprovechar las propiedades de la mecánica cuántica que ofrece un ordenador cuántico, ya que, por ejemplo, Python es uno de los lenguajes tradicionales más utilizados en la computación cuántica, pero no explota las propiedades de este tipo de computación al ser un lenguaje estándar de propósito general.

Algunos de los recursos para programar sobre circuitería cuántica que actualmente existen son Qiskit, Q #, Cirq, Silq, QLib.

- **Qiskit** y **Cirq**: Estos recursos son dos librerías de Python que tienen la capacidad de poder escribir, manipular y optimizar circuitos cuánticos y ejecutarlos sobre ordenadores cuánticos, simuladores de ordenadores cuánticos u ordenadores clásicos. Qiskit es es la librería mas utilizada actualmente en el mercado ya que esta puede aprovechar en gran medida todas las propiedades de la computación cuántica.

- **Q #**: Q # es un lenguaje de programación de código abierto de *Microsoft* para desarrollar y ejecutar algoritmos cuánticos.

- **Silq**: Silq es un nuevo lenguaje de computación cuántica de alto nivel, diseñado por los informáticos del instituto federal de tecnología *ETH Zurich*. La sintaxis utilizada para escribir código en este lenguaje es parecido al utilizado en los lenguajes actuales [57].

- **QLib**: QLib es una librería de Matlab, destinada principalmente a usuarios de matemáticas, física y química, quienes utilizan en mayor medida este lenguaje.

A continuación, puede observarse un trozo de código programado con la librería Qiskit [1]. Este código no contiene ninguna implementación a comentar, únicamente se muestra el código para visualizar como se realiza la programación con esta librería. Como se puede observar, la programación se realiza a nivel de las puertas lógicas, es decir, se interactúa directamente con el funcionamiento de los cúbits sobre las puertas lógicas cuánticas y no como comúnmente se realiza la programación a alto nivel. Más adelante, en el capítulo 8, se muestran unos ejemplos en detalle con esta librería.

```
# qc: El circuito cuántico con el que se va trabajar
# cubits: El número de cúbits que se van a utilizar

def Inicializar (qc, cubits):
    for q in cubits:
        qc.h(q) # Se aplica la puerta de Hadamard sobre cada cúbit para inicializarlos
    return qc # Se devuelve el circuito cuántico

....

qc.cx(0,1) # Se aplica una puerta al circuito cuántico
qc.measure(1,0) # Se realiza la medición en el circuito
qc.draw() # Se dibuja el circuito y se muestra por pantalla
```

Código 5.1: Ejemplo Qiskit.

Por otra parte, el código que se realiza mediante el lenguaje Silq se parece más al que normalmente se suele utilizar. A continuación se muestra un fragmento de código para visualizar la sintaxis de este [57]. Este código no contiene ninguna implementación a comentar, únicamente se muestra el código para visualizar como se realiza la programación con este lenguaje.

```

def solve[n:!N](bits !B^n){
    x := H(0:B); # Preparar superposición entre 0 y 1
    qs := if x then bits else (0:int[n]) as B^n; # Preparar superposición entre 'bits' ↩
```

```
    ↪ y 0
    forget(x=qs[0]); # No computar 'x' si 'bits[0]==1'
    return qs;
}

#Llamada a la función con 'bits=1' y 'n=2'
def main(){
    x := 1:!int[2];
    y := x as !B^2;
    return solve(y);
}
```

Código 5.2: Ejemplo de Silq.

Como se puede observar, el lenguaje se asemeja en gran medida a la programación actual. Sin embargo, aún falta recorrido para conseguir obtener la legibilidad en el código con la que contamos actualmente.

Gran parte de los encargados de realizar nuevos algoritmos son matemáticos, físicos e informáticos trabajando en el campo de *Computer Science* ya que son los adecuados por sus conocimientos en el ámbito necesario.

6 Computación cuántica en la algorítmia

Un algoritmo es un conjunto de instrucciones ordenadas y finitas que permite solucionar un problema mediante el procesamiento de un dispositivo. En teoría, un algoritmo cuántico tiene el mismo objetivo, solucionar un problema en un tiempo determinado.

La diferencia entre un algoritmo clásico y un algoritmo cuántico es el diseño del algoritmo debido a que se necesita estructurar el algoritmo de forma diferente. Actualmente, con las opciones que se tienen, los algoritmos cuánticos trabajan directamente sobre las puertas cuánticas pero mediante instrucciones del lenguaje de programación, es decir, se establece en las instrucciones que puertas cuánticas se van a ir utilizando en el algoritmo. Por otra parte, los algoritmos clásicos trabajan con instrucciones mas entendibles por el programador y no trabajan directamente sobre el funcionamiento de las puertas cuánticas para realizar el algoritmo.

Los algoritmos cuánticos están diseñados para trabajar con cúbits y aprovechar sus propiedades, pero los algoritmos clásicos están diseñados para trabajar con bits. Por lo tanto, el tiempo de ejecución de un algoritmo cuántico siempre será mucho menor que el tiempo del mismo algoritmo en la computación clásica. De la misma forma, aunque se utilicen librerías como Qiskit (librería comentada en el capítulo 6) para ejecutar código, en este caso, sobre simuladores de ordenadores cuánticos, no se obtendría todo el potencial que ofrece la computación cuántica ya que no se esta ejecutando realmente sobre un ordenador cuántico sino sobre una simulación. Esto ocurre debido a todas las propiedades de la computación cuántica y de los cúbits, comentadas en el capítulo 2.

6.1 Algoritmo cuántico

Un algoritmo cuántico se ejecuta en un circuito cuántico mediante la computación cuántica. En la **Teoría de la complejidad computacional**, *Bounded-probability Quantum Polynomial* (BQP) representa la clase de algoritmos que pueden ser resueltos en un ordenador cuántico en tiempo polinómico con un margen de error promedio inferior a $\frac{1}{4}$.

En el análisis de los algoritmos cuánticos es habitual comparar la cota superior asintótica con el mejor algoritmo clásico conocido, o si el problema está resuelto, con el mejor algoritmo clásico posible.

6.1.1 Teoría de la complejidad cuántica

La teoría de complejidad cuántica forma parte de la teoría de la complejidad computacional. Esta se ocupa de la dureza de los problemas en relación a las clases de complejidad y la relación entre clases de complejidad cuántica y clases de complejidad clásica.

Una clase de complejidad es la colección de problemas computacionales que pueden ser resueltos bajo unas ciertas restricciones en cuanto a los recursos. Por ejemplo, la clase de complejidad clásica "P" define el conjunto de problemas que una máquina de *Turing* puede resolver en tiempo polinomial. De igual forma, las clases de complejidad cuánticas se pueden definir como modelos cuánticos de computación, como seria la máquina cuántica de *Turing*.

Una de las razones por la que se estudia la teoría de la complejidad cuántica es debido a la tesis de ***Church-Turing*** [58].

La tesis de *Church-Turing* establece que cualquier modelo computacional puede simularse en tiempo polinomial con una máquina de *Turing* probabilística, pero aún no se puede afirmar si es válido para un contexto de computación cuántica. Es muy probable que una máquina probabilística de *Turing* no pueda simular modelos de computación cuántica en tiempo polinomial.

Es por esto que contamos con la máquina de *Turing* cuántica, la cual puede resolver problemas en el contexto de computación cuántica. Además, como se ha comentado anteriormente, si esta clase de problemas son resueltos de forma eficiente con un error acotado concreto, a esta clase de problemas o algoritmos se les llama algoritmos BQP.

6.2 Algoritmos cuánticos fundamentales

Se van a comentar 3 algoritmos cuánticos que actualmente existen y son relevantes para la evolución de la computación cuántico debido a que su diseño abre las puertas a otros posibles desarrollos y funcionalidades en la algoritmia cuántica. Algunos de estos algoritmos ya se han ido introduciendo durante el desarrollo del proyecto, como por ejemplo el algoritmo de *Shor* en el capítulo 3 donde se comentaba que este podría ser un gran peligro para la criptografía actual. A continuación se explicarán los 3 algoritmos: **Algoritmo de *Deutsch-Jozsa***, **Algoritmo de *Shor*** y **Algoritmo de *Grover***.

6.2.1 Algoritmo de Deutsch-Jozsa

El algoritmo cuántico de *Deutsch-Jozsa* tiene como objetivo decidir si una función booleana dada es constante o balanceada evaluando la función una única vez. A lo largo de esta sección se explicarán estas dos posibles opciones.

Este algoritmo cuántico fue propuesto por *David Deutsch* y *Richard Jozsa* en 1992, y fue

uno de los primeros pensado para ser ejecutado en un ordenador cuántico [79]. Este tiene el potencial necesario para ser más eficiente que los algoritmos clásicos ya que aprovecha el paralelismo de los estados de superposición cuántica.

El algoritmo de *Deutsh-Jozsa* comienza con una función booleana, cuya formulación matemática se puede representar por la ecuación 6.1. Esta tiene como entrada tantos valores como se desee. El número de entradas de la función no es importante, lo que importa es la cantidad de veces que se evalúa dicha función para determinar la salida, obteniendo en esta 0 ó 1. Por ejemplo, si se tiene una función que recibe como entrada un valor entero y se cuenta con 8 números (0, 1, 2, 3, 4, 5, 6, 7), se debe evaluar 8 veces la función booleana, una por cada número entrante. Esta función booleana podría ser, por ejemplo, determinar si un número es primo o no, y lo que se quiere conocer es si dicha función es constante o balanceada. Por otra parte, gracias al algoritmo cuántico de *Deutsh-Jozsa* solo sería necesario evaluar la función una sola vez.

$$f : \mathbb{R}^n \to \{0, 1\} \tag{6.1}$$

La propia función f determina si el resultado debe ser 1 ó 0. Por ejemplo, si la función fue implementada para saber si un número es primo o no, esta devolverá 1 si el número entrante es primo y 0 si el número entrante no es primo. Es decir, no se requiere ningún número concreto de entradas ni una lógica especifica para la función, únicamente esta función debe ser booleana, tener n entradas y generar una salida, 1 ó 0. El algoritmo no depende de la implementación y el diseño de la función sino del hecho de que la función sea booleana.

Una de las propiedades de este algoritmo es que se garantizará que la función será o balanceada o constante. Una función **constante** devolverá siempre 0 o 1 para todas las entradas, mientras que una función será **balanceada** cuando para la mitad de las entradas la función devuelva 0 y para la otra mitad 1. Por ejemplo, si se cuenta con una función de una única entrada y hay 4 entradas (0,1,2,3) y se evalúan en una función booleana cualquiera, los resultados a tener en cuenta pueden ser alguno de los siguientes casos representados en la tabla 6.1

Tabla 6.1: Función balanceada o constante

f(0)	**f(1)**	**f(2)**	**f(3)**	**Resultado**
0	0	0	0	Constante
0	0	1	1	balanceada
0	1	0	1	balanceada
1	0	0	1	balanceada
0	1	1	0	balanceada
1	0	1	0	balanceada
1	1	0	0	balanceada
1	1	1	1	Constante

El objetivo, como ya se ha comentado, es averiguar si la función es constante o balanceada realizando el menor número de búsquedas posibles, es decir, evaluar lo menos posible para saber si la función es constante o balanceada.

Para este cometido contamos con dos posibles soluciones, una **solución clásica** y la **solución cuántica**. La solución clásica deberá evaluar tantas veces como sea necesario y la solución cuántica solo evaluará una vez todos los valores.

6.2.1.1 Solución clásica

En el **mejor de los casos**, con dos llamadas a la función se puede determinar si la función esta balanceada o no.

En el **peor de los casos**, si se continua obteniendo el mismo resultado para cada entrada que asignamos a la función, se tendrá que comprobar exactamente la mitad de los posibles valores de entrada más uno para asegurarse que f es constante ya que, como se ha comentado, una función es balanceada cuando exactamente la mitad es 0 y la otra mitad es 1. Si se comprueba que la mitad de las entradas más uno su resultado es, por ejemplo, 0 esto quiere decir que la función será constante.

Si se obtiene el mismo resultado continuamente, se puede expresar la probabilidad de que la función sea constante. En la práctica, se puede establecer una cota para determinar con qué probabilidad se desea asegurar el resultado. Si queremos estar seguros al 100%, deberemos comprobar la mitad de los posibles valores más uno.

Como se puede observar, esta solución no es del todo fiable ya que establecer una cota para determinar la probabilidad de un resultado no asegura que el resultado sea el correcto. Si no se establece dicha cota hay que realizar el algoritmo en el peor de los casos por lo que tomaría mucho tiempo de cómputo en comparación, pero sí se aseguraría que el resultado fuera el correcto.

6.2.1.2 Solución cuántica

Con el uso de un ordenador cuántico, este problema se puede resolver con una seguridad del 100% con solo una llamada a la función $f(x)$, siempre que esta función este implementada como un **oráculo cuántico**.

Una función oráculo U es una operación de caja negra que se utiliza como entrada para otro algoritmo. Generalmente, la función oráculo se define por la ecuación 6.2. Esta ecuación representa el bloque U_f de la figura 6.1 donde encontramos unos valores de entrada por x con t cúbits y unos valores de entrada por y con g cúbits. Más adelante se explica el funcionamiento de este bloque.

$$U(|x\rangle \otimes |y\rangle) = |x\rangle \otimes |y \oplus f(x)\rangle \tag{6.2}$$

siendo $x \in \{0,1\}^t$ e $y \in \{0,1\}^g$, siendo t y g una cantidad de cúbits entrada.

En este caso el oráculo cuántico estará construido por una función que tiene de entradas el mismo número que de salidas. Se debe recordar, de la sección 2.3, que los componentes del circuito cuántico tienen siempre el mismo número de entradas que de salidas.

Las entradas y las salidas serán los cúbits, pero los cúbits de salida no tienen porque tener los mismos estados que los cúbits de entrada. El oráculo se encarga de realizar las operaciones correspondientes sobre los cúbits. Las operaciones que realiza dependerán de la funcionalidad de $f(x)$ en la ecuación 6.2. En este caso $f(x)$ será la función booleana que se necesita para el algoritmo cuántico de *Deutsh-Jozsa*.

A partir del circuito 6.1, se puede crear el algoritmo cuántico, teniendo en cuenta que el oráculo esta definido en la imagen como U_f.

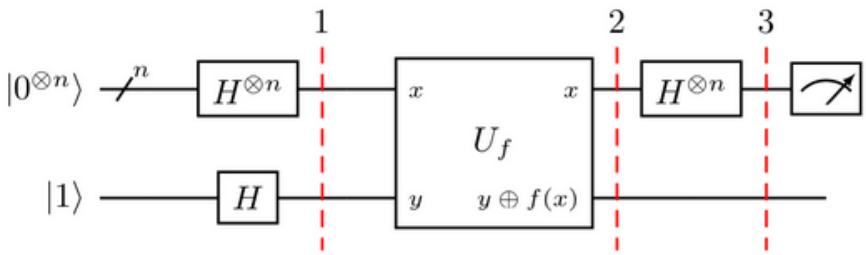


Figura 6.1: Circuito cuántico *Deutsch-Jozsa*. Obtenida de [60].

A continuación, se explican los pasos para el algoritmo:

1. Se prepara un registro de t cúbits inicializado a 0 y un segundo registro de un cúbit inicializado a 1. Este registro se representa por la ecuación 6.3.

$$|\psi_0\rangle = |0\rangle^{\oplus n}|1\rangle \tag{6.3}$$

2. Se aplica la puerta de *Hadamard* (puerta cuántica explicada en la sección 2.3) a cada uno de los cúbits. Para ello utilizamos el sumatorio en *Hadamard* para realizarlo sobre cada uno de los t cúbits a 0, y sobre un único cúbit a 1. Matemáticamente, esto se define en la ecuación 6.4.

$$|\psi_1\rangle = \frac{1}{\sqrt{2^{n+1}}} \sum_{x=0}^{2^n-1} |x\rangle(|0\rangle - |1\rangle) \tag{6.4}$$

3. Se aplica el oráculo cuántico para obtener los nuevos estados a partir de los obtenidos de las puertas de *Hadamard*. La aplicación del oráculo sobre los cúbits se representa por la ecuación 6.5.

$$|\psi_2\rangle = \frac{1}{\sqrt{2^{n+1}}} \sum_{x=0}^{2^n-1} |x\rangle(|f(x)\rangle - |1 \oplus f(x)\rangle) = \frac{1}{\sqrt{2^{n+1}}} \sum_{x=0}^{2^n-1} (-1)^{f(x)}|x\rangle(|0\rangle - |1\rangle) \quad (6.5)$$

4. Se aplica la puerta de *Hadamard* para cada cúbit del primer registro. Matemáticamente, este registro se define en la ecuación 6.6.

$$|\psi_3\rangle = \frac{1}{2^n} \sum_{x=0}^{2^n-1} (-1)^{f(x)} \left[\sum_{y=0}^{2^n-1} (-1)^{x \cdot y} |y\rangle \right] = \frac{1}{2^n} \sum_{y=0}^{2^n-1} \left[\sum_{x=0}^{2^n-1} (-1)^{f(x)} (-1)^{x \cdot y} \right] |y\rangle \quad (6.6)$$

5. Por último, se mide el primer registro de los t cúbits a partir de la función de probabilidad. Esto queda representado matemáticamente por la ecuación 6.7.

$$|0\rangle^{\oplus n} = \left| \frac{1}{2^n} \sum_{x=0}^{2^n-1} (-1)^{f(x)} \right|^2 \quad (6.7)$$

   La cual se evalúa a 1 si $f(x)$ es constante y a 0 si $f(x)$ es balanceada.

Por lo tanto, tras estos pasos, el rendimiento con una solución cuántica es mucho mayor gracias a la reducción tan drástica del número de llamadas que hacemos a la función en comparación a la solución clásica.

6.2.2 Algoritmo de Shor

El algoritmo de *Shor* es un algoritmo que permite encontrar factores de un número entero N en tiempo polinomial. Este algoritmo se introdujo en el capítulo 3 haciendo hincapié en el problema que supondría para la criptografía de clave pública, como sería con RSA.

Hasta el momento no se ha podido romper algoritmos tan fuertes como RSA, pero el algoritmo de *Shor* tiene el potencial para hacerlo. En 2001 IBM utilizó el algoritmo cuántico de *Shor* para descomponer el número 15 en sus factores usando un ordenador cuántico con 7 cúbits [9]. Aunque el número 15 sea un número bajo hay que tener en cuenta que 7 cúbits también son poca cantidad. Tener mayor cantidad de cúbits en un entorno cuántico nos permitiría factorizar con números mas elevados. Cabe recordar del capítulo 2 que tener un

cúbit más no es como tener un bit más, la capacidad de resultados que se pueden tener con los cúbits es de 2^n, siendo n los cúbits a utilizar.

En otras palabras, el algoritmo de *Shor* podrá romper RSA en tiempo polinómico en cuanto este pueda implementarse sobre un ordenador cuántico real con los cúbits suficientes para realizarlo. En este caso, estamos contemplando solo RSA, pero toda la criptografía se vería comprometida.

El algoritmo de *Shor* consta de dos partes:

- Una reducción del costo de descomposición en factores primos al problema de encontrar el periodo, pudiéndose hacer esta parte con computación clásica.
- Algoritmo cuántico para solucionar el problema de encontrar el periodo.

6.2.2.1 Parte clásica

1. Se escoge un número aleatorio: $1 < a < N$, siendo a y N cualquier numero entero positivo. Se puede escoger cualquier número entero pero hay que considerar que para números muy grandes se necesitarán también mas cúbits.
2. Se computa el máximo común divisor de a y N mediante $K = mcd(a, N)$, pudiéndose utilizar el **algoritmo de *Euclides*** [41].
   2.1 Si $K \neq 1$, entonces K es un factor no trivial de N, por lo tanto se termina el cómputo.
   2.2 Si $K = 1$, se utiliza la **parte cuántica** que encontramos en la sección 6.2.2.2 para encontrar r, el período de esta función.
   $$f(x) = a^x \pmod N \tag{6.8}$$
   r sería el número entero positivo mas pequeño que cumpliese $a^r \equiv 1 \pmod N$.
3. Si r es impar o $a^{\frac{r}{2}} \equiv -1 \pmod N$, volvemos de nuevo al paso 1
4. En otro caso, tanto $mcd(a^{\frac{r}{2}} + 1, N)$ como $mcd(a^{\frac{r}{2}} - 1, N)$ son factores no triviales de N, por lo tanto se termina el algoritmo.

6.2.2.2 Parte cuántica: Subprograma para encontrar el periodo

Los circuitos cuánticos usados para este algoritmo están diseñados a medida para cada elección de números enteros N y para cada elección aleatoria de a usada en la ecuación 6.8.

Dado N, se debe encontrar un valor $Q = 2^q$ tal que $N^2 \leq Q \leq 2N^2$, lo que implica que

$\frac{Q}{r} > N$. Los registros de cúbit de entrada y salida necesitan contener superposiciones de valores de 0 a $Q-1$ y q cúbits cada uno.

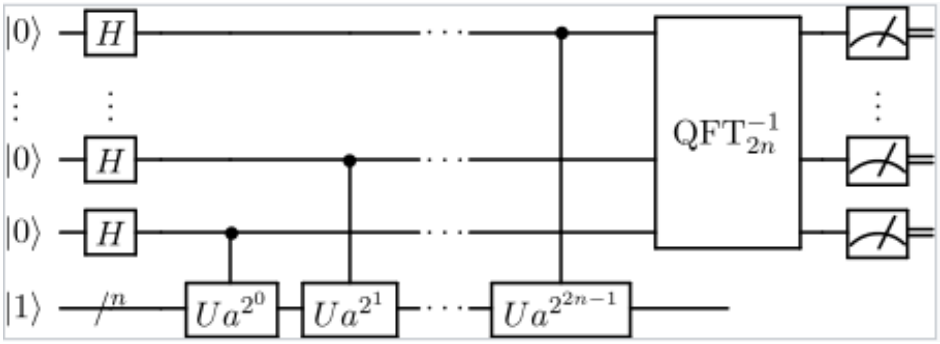


Figura 6.2: Subprograma para encontrar el periodo. Obtenida de [60].

Este subprograma, representado por la figura 6.2, sigue los siguientes pasos:

1. Se realiza la inicialización de los registros. Para ello se aplican los estados 0 de los cúbits y luego se aplica la puerta de *Hadamard* en paralelo a cada uno de los q cúbits, donde $Q = 2^q$. Este proceso se puede ver reflejado en la ecuación 6.9.

$$\frac{1}{\sqrt{Q}} \sum_{x=0}^{Q-1} |x\rangle = \left(\frac{1}{\sqrt{2}} \sum_{x_1=0}^{1} |x_1\rangle \right) \otimes \ \ldots \ \otimes \left(\frac{1}{\sqrt{2}} \sum_{x_q=0}^{1} |x_q\rangle \right) \tag{6.9}$$

2. Se construye $f(x)$ como una función cuántica y la aplicamos sobre el estado anterior. En la ecuación 6.10 se puede ver reflejado el proceso de construcción de la función cuántica.

$$U_f|x, 0^q\rangle = |x, f(x)\rangle \rightarrow U_f \frac{1}{\sqrt{Q}} \sum_{x=0}^{Q-1} |x, 0^q\rangle = \frac{1}{\sqrt{Q}} \sum_{x=0}^{Q-1} |x, f(x)\rangle \tag{6.10}$$

   El valor de r que estamos buscando se almacena en la fase de los cúbits de entrada x como el resultado del *retroceso de fase*.

3. Se aplica la inversa de la **transformada cuántica de Fourier** o *Quantum Fourier Transform* (QFT) al registro de entrada [47]. Esta transformación usa la Q-ésima raíz de la unidad para distribuir la amplitud de cualquier $|x\rangle$ estado por igual entre todos los Q de los $|y\rangle$ estados, haciéndolo de manera diferente para cada x.

$$U_{QFT}(|x\rangle) = \frac{1}{\sqrt{Q}} \sum_{y=0}^{Q-1} \omega^{xy} |y\rangle, \tag{6.11}$$

siendo $\omega = e^{\frac{2\pi i}{Q}}$.

Conduciendo esto al estado final:

$$\frac{1}{Q}\sum_{x=0}^{Q-1}\sum_{y=0}^{Q-1}\omega^{xy}|y, f(x)\rangle \tag{6.12}$$

Y siendo esta la suma reordenada del estado final:

$$\frac{1}{Q}\sum_{x=0}^{Q-1}\sum_{y=0}^{Q-1}\left[\sum_{x\in\{0,\ldots,Q-1\};f(x)=z}\omega^{xy}\right]|y, z\rangle \tag{6.13}$$

4. Se realiza una medición (propiedad cuántica comentada en la sección 2.2.1) y se obtiene un valor de y en la entrada y un resultado z en la salida. Al ser f periódica, la probabilidad de medir algún $|y, z\rangle$ viene dado por las ecuaciones 6.14 y 6.15.

$$P_r(|y, z\rangle) = \left|\frac{1}{Q}\sum_{x\in\{0,\ldots,Q-1\};f(x)=z}\omega^{xy}\right|^2 = \frac{1}{Q^2}\left|\sum_{b=0}^{m-1}\omega^{(x_0+rb)y}\right|^2 = \frac{1}{Q^2}|\omega^{x_0y}|^2\left|\sum_{b=0}^{m-1}\omega^{bry}\right|^2 \tag{6.14}$$

$$\frac{1}{Q^2}|\omega^{x_0y}|^2\left|\sum_{b=0}^{m-1}\omega^{bry}\right|^2 = \frac{1}{Q^2}\left|\sum_{b=0}^{m-1}\omega^{bry}\right|^2 = \frac{1}{Q^2}\left|\frac{w^{mry}-1}{w^{ry}-1}\right|^2 = \frac{1}{Q^2}\frac{\sin^2\frac{\pi mry}{Q}}{\sin^2\frac{\pi ry}{Q}} \tag{6.15}$$

Esta probabilidad es mayor cuanto más cerca está el vector W^{ry} (este vector puede encontrarse en la ecuación 6.15) del eje real positivo o cuanto más cerca esta $\frac{yr}{Q}$ de un número entero. A menos que r sea una potencia de 2, no será un factor de Q.

5. Dado que $\frac{y\cdot r}{Q}$ está cerca de algún número entero c, el valor conocido $\frac{y}{Q}$ está cerca del valor conocido $\frac{c}{r}$. Si realizamos una expansión de la fracción continua sobre $\frac{y}{Q}$ nos permite encontrar aproximaciones $\frac{d}{s}$ que satisfacen las condiciones siguientes:

   - $s < N$
   - $\left|\frac{y}{Q} - \frac{d}{s}\right| < \frac{1}{2Q}$

   Asumiendo que $\frac{d}{s}$ es irreducible junto con estas condiciones, s es muy probable que sea el periodo indicado r, o al menos un factor de este.

6. Se comprueba la ecuación 6.16 y si se cumple entonces se finaliza el algoritmo.

$$f(x) = f(x+s) \iff a^s \equiv 1 \pmod{N} \tag{6.16}$$

7. Si no se cumple la ecuación 6.16, se deben obtener más candidatos de r usando múltiplos de s o usando otras s con $\frac{d}{s}$ cerca de $\frac{y}{q}$. Si algún candidato funciona, entonces se finaliza.

8. Si no funciona ni el paso 6 ni el paso 7, comenzamos de nuevo en el paso 1 del subprograma. Al repetir el subprograma, la medición de los cúbits no tiene porque ser la misma por lo que los resultados pueden cambiar.

Este algoritmo se ve implementado más adelante, en la sección 8.2.

6.2.3 Algoritmo de Grover

El algoritmo de *Grover*, también conocido como algoritmo de búsqueda cuántica, fue ideado por *Lov Grover* en 1996 [89]. Este algoritmo cuántico realiza una búsqueda no estructurada que encuentra con alta probabilidad la entrada única a una función de caja negra que produce una salida concreta. La complejidad de este algoritmo en cómputo cuántico es de $O(\sqrt{N})$ y con computación clásica de $O(N)$, siendo N el número de entradas.

Casi al mismo tiempo de la publicación del algoritmo, otros investigadores[1] demostraron que cualquier solución cuántica a este problema requiere de evaluar la función $\Omega(\sqrt{N})$ veces, por lo que significa que el algoritmo de *Grover* es **asintóticamente óptimo** [46]. Es decir, para la mayoría de las entradas que se realizan en el algoritmo, en el peor de los casos, se produce un resultado constante.

Muchos otros algoritmos cuánticos pueden proporcionar una aceleración exponencial sobre sus homólogos clásicos, pero el algoritmo de *Grover* solo proporciona una aceleración cuadrática. Sin embargo, esta aceleración cuadrática es notoria cuando N es grande. El algoritmo de *Grover* podría mediante fuerza bruta obtener una clave criptográfica simétrica de 128 bits en 2^{64} iteraciones o una de 256 bits en 2^{128} iteraciones. Es decir, que para este algoritmo de *Grover* el coste computacional seria de $O(2^{\frac{n}{2}})$. Este coste es mucho menor al obtenido con la computación clásica, $O(2^n)$, pero aún así no es suficiente como para comprometer la seguridad de AES a corto plazo [46] [68].

Cualquier tarea de búsqueda, se puede formular con una función abstracta $f(x)$ aceptando elementos de búsqueda x. Si el elemento x es una solución para la tarea de búsqueda, entonces $f(x) = 1$, sino $f(x) = 0$. El problema de búsqueda consiste en buscar un x_0 tal que $f(x_0) = 1$. Además, se asume que solo un índice satisface $f(x_0) = 1$ y a este se le llama ω. El objetivo es identificar dicho ω.

A esta función f podemos llegar mediante un oráculo, anteriormente explicado en la sección

[1] *Charles H. Bennett , Ethan Bernstein, Gilles Brassard y Umesh Vazirani.*

6.2.1.2, en forma de operador unitario U_w (Ecuación 6.17)

$$\begin{cases} U_\omega|x\rangle = -|x\rangle & \text{para } x = \omega, \text{ siendo } f(x) = 1, \\ U_\omega|x\rangle = |x\rangle & \text{para } x \neq \omega, \text{ siendo } f(x) = 0 \end{cases} \tag{6.17}$$

Este algoritmo genera ω con una probabilidad de al menos 0.5 usando $O(\sqrt{N})$ aplicaciones de U_ω.

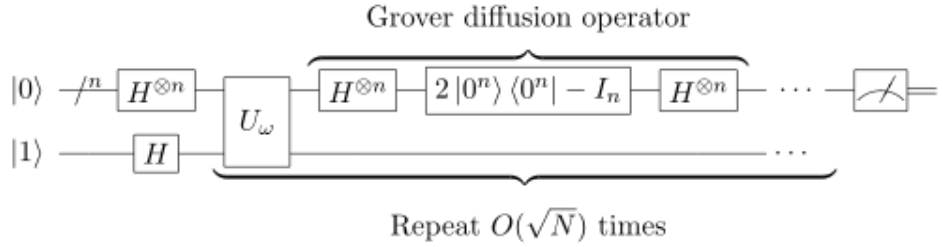


Figura 6.3: Circuito cuántico de *Grover*. Obtenida de [89].

Pasos para realizar el algoritmo de *Grover*, representado en la figura 6.3:

1. Se inicializa el sistema mediante la puerta de *Hadamard*, aplicando superposición a todos los cúbits entrantes. La aplicación de la superposición a los cúbits se representa mediante la ecuación 6.18.

$$|s\rangle = \frac{1}{\sqrt{N}} \sum^{N-1} x = 0|x\rangle \tag{6.18}$$

2. Se realizan $r(N)$ **iteraciones de *Grover***:

   2.1 Se aplica el operador U_ω, representado en la ecuación 6.17

   2.2 Se aplica el operador de difusión de *Grover*, representado por la ecuación 6.19.

$$U_s = 2\,|s\rangle\,\langle s| - I, \tag{6.19}$$

   siendo $|s\rangle\langle s|$ una multiplicación de matrices e I la matriz identidad.

3. Se mide el estado cuántico resultante.

Este algoritmo se ve implementado más adelante, en la sección 8.1.

7 Computación cuántica en inteligencia artificial

La inteligencia artificial es el campo de la ciencia que estudia y abarca un conjunto extenso de algoritmos, técnicas y estrategias, desde algoritmos de lógica difusa, hasta técnicas más modernas de aprendizaje automático o incluso aprendizaje profundo, que intentan proveer a las máquinas la capacidad de tomar decisiones de forma autónoma para resolver problemas concretos con la menor intervención humana posible. Existen tres tipologías de inteligencia artificial clasificadas según el número y tipo de acciones que un procesador de información puede realizar.

1. **Inteligencia artificial débil**: Donde se realizan funciones como procesamiento del lenguaje, reconocimiento de imágenes y sonidos, traductores, autoconducción, etc...
2. **Inteligencia artificial general (fuerte o de nivel humano)**: Donde las máquinas pueden realizar cualquier actividad intelectual, de forma análoga a los seres humanos.
3. **Superinteligencia**: Se destaca por superar a la inteligencia humana en todos los ámbitos.

Actualmente nos encontramos en desarrollo de la inteligencia artificial débil ya que las otras son suposiciones e ideas que se desean alcanzar.

Muchos sistemas de inteligencia artificial destacan por aprender de los datos que van recibiendo, es decir, aprenden de la "experiencia". Por ejemplo, las redes neuronales pueden aprender de la experiencia debido a que estas reciben datos de aprendizaje con los que forman patrones para poder responder con mas eficacia a los problemas a los que se van enfrentando. Estos sistemas se consiguen con las técnicas del **aprendizaje automático** (*Machine Learning*). Estas técnicas no son las únicas para hacer que una máquina tome decisiones, pero sí son las que proporcionan soluciones más generalizables y por tanto son los más utilizadas. Cabe destacar que dentro del aprendizaje automático existen técnicas muy utilizadas como el **aprendizaje profundo** (*Deep Learning*), el cual es de gran utilidad en el aprendizaje para las redes neuronales.

En la sección 7.3 se podrá observar como la computación cuántica puede ayudar en gran medida a estas técnicas que, a continuación, se van a comentar.

7.1 Aprendizaje automático

El aprendizaje automático es una rama del campo de la Inteligencia Artificial que agrupa una serie de algoritmos que intentan resolver problemas de predicción y clasificación. Además permite a los algoritmos identificar patrones complejos utilizando una cantidad significativa de datos.

Como ya se ha comentado, el aprendizaje automático agrupa un conjunto de algoritmos que intenta resolver problemas concretos, pero para que estos algoritmos consigan aprender se cuenta con diferentes tipos de aprendizaje que el aprendizaje automático puede utilizar. A continuación se van a comentar los tipos de aprendizaje.

- **Aprendizaje supervisado**: Se entrena a los sistemas con datos etiquetados. Los datos etiquetados son una designación de los datos que se han etiquetado con una o más etiquetas que identifican ciertas propiedades o características, o clasificaciones u objetos contenidos. Inicialmente se entrena el sistema con un conjunto de datos de referencia o de "entrenamiento", para generar un modelo que represente esos datos y resuelva la tarea para la que se ha preparado. Después este modelo se usará para procesar nuevos datos, que no están etiquetados, para obtener un resultado. De esta forma, cuando se le den los datos no etiquetados al algoritmo ya entrenado, este será capaz de realizar las pertinentes acciones con respecto a la "experiencia" que ha conseguido "trabajando" con los datos etiquetados.

- **Aprendizaje no supervisado**: En este caso, no se cuenta con un conocimiento previo, el sistema observa características o comportamientos en los datos y busca similitudes y patrones para aprender de los datos reales.

- **Aprendizaje semi-supervisado**: Este tipo de aprendizaje es un aprendizaje supervisado donde los datos contienen pocos ejemplos etiquetados en comparación al aprendizaje supervisado. Este aprendizaje se realiza debido a que el coste computacional del etiquetado en los datos suele ser elevado y se intenta disminuir dicho coste generando menos etiquetado en los datos.

- **Aprendizaje por refuerzo**: El sistema aprende a partir de su propia experiencia, en base al proceso de prueba y error.

7.2 Aprendizaje profundo

El aprendizaje profundo es un modelo de aprendizaje por capas. Con este modelo se procesa la información en varias etapas para poder tener en cuenta interacciones complejas entre los datos. El aprendizaje profundo se basa en el uso de **redes neuronales**, sistemas que se inspiran en el funcionamiento de las neuronas de un cerebro biológico, por tanto, intentan aproximarse a nuestra forma de aprender. Cuanto más profunda sea la red, es decir, cuantas

más capas tenga, más se podrá adaptar esta a los datos que se reciban del entrenamiento ya que la red podrá extraer más patrones y características. Sin embargo, hay una limitación en la cantidad de capas a utilizar ya que cuanto más profunda sea la red (mas capas se tengan) mas se adaptará esta a los datos de entrenamiento, por lo que si se utilizan datos que cambien un poco a los del entrenamiento, el sistema fallará con más regularidad. Este fenómeno es conocido como *overfitting*.

En una red neuronal se cuenta con diferentes capas. Inicialmente se tiene una primera capa, la capa de aprendizaje, en la cual las neuronas iniciales procesan los datos de entrada individualmente. Posteriormente se cuenta con una segunda capa de aprendizaje donde los resultados de la capa anterior se tratan para tomar una decisión mas compleja. Por último, se van añadiendo mas capas de aprendizaje las cuales manejarán los resultados de la capa anterior. El motivo de añadir mas capas es para poder extraer más patrones y características, como ya se ha comentado anteriormente.

El proceso de entrenamiento de la red no suele ser rápido y una gran cantidad de datos podría ralentizar este proceso.

7.3 Inteligencia artificial cuántica

Las técnicas modernas de inteligencia artificial utilizan una gran cantidad de datos para que los sistemas aprendan y después puedan realizar las labores demandadas. A medida que el problema cuenta con mayor cantidad de datos, el trabajo que hace el sistema es mayor y por tanto también es mayor el tiempo que tarda en procesar dichos datos, el de encontrar patrones y el de aprender.

Como ya se ha visto, los ordenadores cuánticos están diseñados para realizar cálculos de forma más eficiente y precisa que los ordenadores clásicos. Dadas las condiciones actuales, el progreso de la inteligencia artificial **depende de los avances en computación cuántica**. Para los expertos en este campo, la computación cuántica representará el salto cuantitativo y cualitativo que la inteligencia artificial necesita [28].

Con la unión de la inteligencia artificial y la computación cuántica surge el término de **inteligencia artificial cuántica**.

El aprendizaje automático clásico resuelve problemas de clasificación y regresión con algoritmos cuya velocidad de cálculo está limitada si el espacio de características es grande. Esta unión de inteligencia artificial y computación cuántica ayudaría a eliminar esta limitación.

Supongamos que contamos con una inteligencia artificial que obtenga los datos de todos los usuarios de internet. Actualmente, cada día, hay 3200 millones de usuarios que utilizan internet en el mundo produciendo alrededor de 2.5 exabytes de datos, es decir, hay una enorme cantidad de datos que se va a procesar [44]. Esto implica que el algoritmo de inteligencia artificial trabajará analizando todos estos datos, pero debido a esta cantidad, conllevaría

mucho tiempo de procesamiento para entrenar un modelo capaz de modelar dichos datos.

Con esto se desea enfatizar que la combinación de inteligencia artificial y computación cuántica puede ser revolucionario para el aprendizaje de la inteligencia artificial.

7.3.1 Aprendizaje automático cuántico

Esta disciplina combina la mecánica cuántica con el aprendizaje automático. Los algoritmos o modelos de aprendizaje automático cuántico utilizan las ventajas de la información cuántica con el fin de mejorar el aprendizaje automático clásico.

Las técnicas de aprendizaje automático clásico utilizan las matemáticas para identificar patrones en conjuntos de datos siendo de gran utilidad en campos de la biomedicina y la física aplicada, entre otros. Con el uso del aprendizaje automático cuántico se puede conseguir agilizar el proceso de aprendizaje en estos campos donde la inteligencia artificial es fundamental y se procesan gran cantidad de datos.

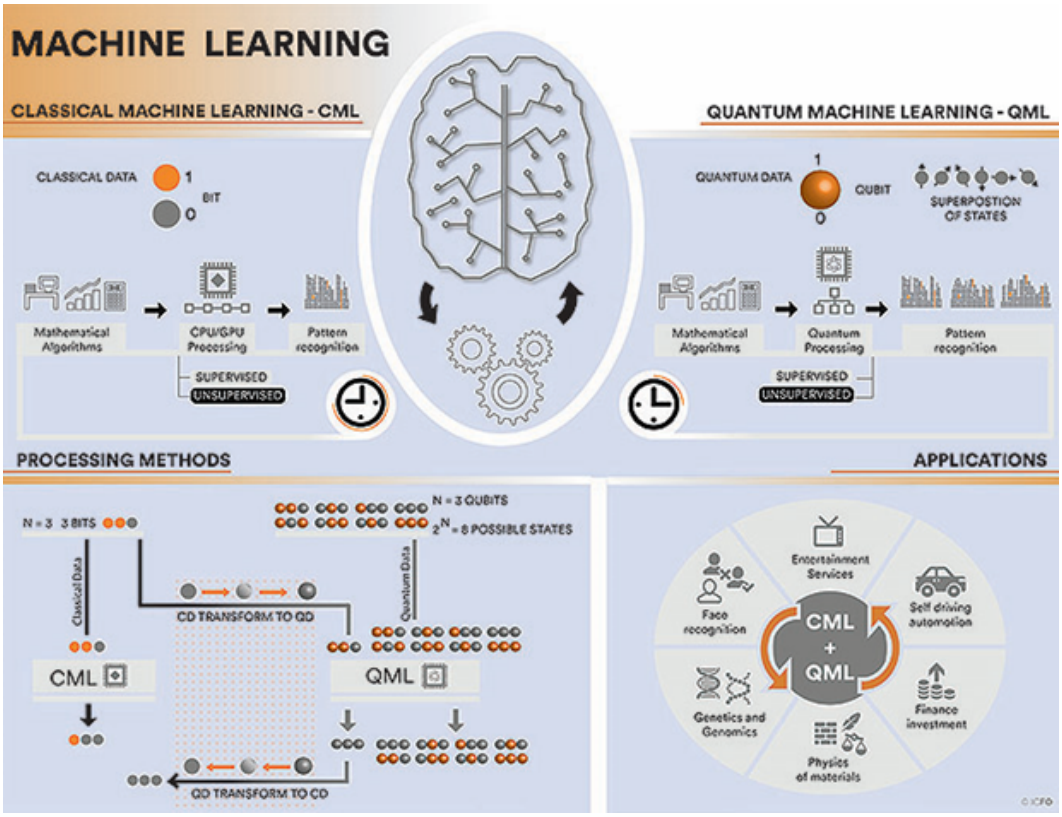


Figura 7.1: Aprendizaje automático clásico y cuántico. Obtenida de [65].

Un estudio publicado en *Nature* por el *Instituto de Ciencias Fotónicas* ha conseguido una revisión del estado actual y real del aprendizaje automático clásico y el cuántico [65]. En este

estudio se ha investigado sobre el método convencional de usar el aprendizaje automático clásico para analizar datos clásicos, sobre el uso del aprendizaje automático cuántico para analizar datos clásicos y cuánticos y sobre el uso del aprendizaje automático clásico para analizar datos cuánticos. Estos métodos se pueden observar en la figura 7.1.

Los científicos buscaron obtener una visión detallada de la situación actual de los protocolos de aprendizaje supervisados y no supervisados en el aprendizaje automático clásico. Seguidamente se introdujo el aprendizaje automático cuántico y fue un acercamiento extenso de cómo esta técnica podría ser utilizada para analizar datos clásicos y cuánticos, teniendo en cuenta que las máquinas cuánticas podrían acelerar el procesamiento gracias al uso de los **temples cuánticos** y al uso de **ordenadores cuánticos**.

El aprendizaje automático cuántico actualmente no es solo una idea o una propuesta, existen ya ejemplos de como se podría implementar. Sin ir mas lejos, existe una implementación en Qiskit (librería de Python de circuitería cuántica comentada en el capitulo 6) donde se crea una **red neuronal híbrida cuántica-clásica** [62].

La figura 7.2 representa el marco que se desea construir. En esta red neuronal contamos con una primera capa clásica donde se reciben los datos. Seguidamente estos datos pasan a una capa cuántica que se encargará de favorecer la velocidad de procesamiento de los datos. Finalmente la última capa genera el resultado.

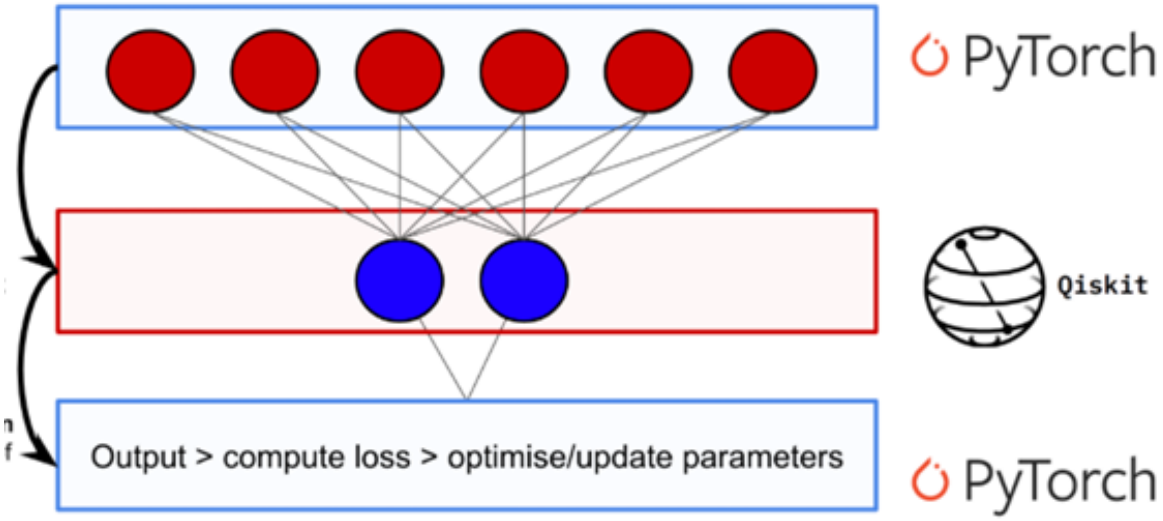


Figura 7.2: Red neuronal híbrida cuántica-clásica. Obtenida de [62].

La capa cuántica en este experimento no genera un gran beneficio ya que no se realiza entrelazamiento cuántico (propiedad cuántica vista en la sección 2.2.2) entre los cúbits. Para obtener una ventaja cuántica con esta capa se deberá mejorar esta capa con el objetivo de hacerla mas sofisticada. El objetivo de este experimento es hacer entender la posible integración de técnicas de *machine learning* y computación cuántica. El experimento puede verse detalladamente en [62].

7.3.1.1 De clasificadores clásicos a clasificadores cuánticos

Actualmente, el clasificador clásico realiza la tarea de clasificación en una tarea de aprendizaje automático, el cual su objetivo es inferir en las etiquetas de clase $y_1, y_2, ..., y_n$ de determinados datos dados. El conjunto de datos de entrenamiento es una colección de datos $D = (x, y)$ donde x son los datos y la y son las etiquetas de entrenamiento

Para comprender bien lo que realiza un clasificador se va a exponer un ejemplo. Imaginemos que se presenta un problema de detección de objetos donde se tienen que clasificar imágenes sobre perros y gatos. El clasificador "observa" cada imagen x (el dato) y si esa imagen es de un perro o un gato y (el etiquetado del dato). Este clasificador valora y "clasifica" la imagen según el etiquetado. Si la imagen es de un perro se agrupará con el resto de imágenes de perro y si es un gato se agrupará con el resto de imágenes de un gato.

Los clasificadores cuánticos son una alternativa eficaz a los clasificadores clásicos debido a su codificación cuántica y el procesamiento de datos de este.

El clasificador cuántico, es una solución cuántica que combina la codificación de datos con un circuito cuántico que se entrelaza y desentrelaza rápidamente seguido de la medida para inferir etiquetas de clase de muestra de datos. Los clasificadores cuánticos nos permiten codificar datos en registros cuánticos, gracias al uso del entrelazamiento cuántico como recurso computacional y al uso de la medida cuántica para la inferencia de datos.

El objetivo es garantizar la caracterización clásica y el almacenamiento de circuitos de sujeto, así como el entrenamiento cuántico/clásico híbrido de los parámetros del circuito, incluso para espacios de características extremadamente grandes.

7.3.2 Redes neuronales cuánticas

Las redes neuronales cuánticas o *Quantum Neural Network* (QNN) son un modelo de red neuronal basado en la mecánica cuántica.

Las primeras ideas sobre esta disciplina se publicaron en 1995 por Subhash Kak y Ron Chrisley, los cuales afirman el uso de la teoría de la **mente cuántica** que postula que los efectos cuánticos juegan un papel en la función cognitiva. La esperanza con las redes neuronales cuánticas es que las características de la computación cuántica, como el **paralelismo cuántico**, los efectos de la **interferencia** y el **entrelazamiento** se puedan aprovechar en estas. Sin embargo, actualmente, al estar en una etapa prematura, los modelos de redes neuronales cuánticos son en su mayoría propuestas teóricas [74].

La mayoría de estas redes neuronales cuánticas se desarrollan como redes *feed-forward*. Esta estructura toma la entrada de una capa de cúbits y transmite esa entrada a otra capa de cúbits. Esta capa de cúbits evalúa la información y envía el resultado a la siguiente capa. Este procedimiento continua hasta la capa final. Estas capas no tienen por qué tener el mismo tamaño, por lo que no tienen por qué tener el mismo número de cúbits que la capa anterior

o posterior.

Los investigadores en esta disciplina han intentado generalizar las redes neuronales al entorno cuántico. Una forma de construir una neurona cuántica es generalizando las neuronas clásicas hasta conseguir hacer puertas unitarias. Las interacciones entre neuronas se pueden controlar cuánticamente. Esto conduce a construir diferentes tipos de redes y diferentes implementaciones de neuronas cuánticas.

Entonces, el primer paso es poder conseguir obtener una representación de las neuronas cuánticas. Una explicación para esto es **la interpretación multimundo**[1] de *Hugh Everett* [8].

Esta teoría proporciona información sobre cómo deben comportarse las redes neuronales cuánticas. Así como las redes neuronales tradicionales están inspiradas en las redes neuronales biológicas, las redes neuronales cuánticas también pueden estar inspiradas en la física cuántica. Investigadores de la Universidad Estatal de Pensilvania utilizaron esta interpretación para desarrollar un método para construir una red neuronal cuántica [8].

Huge Everett establece que hay muchos universos paralelos que contienen muchas realidades. "¿Y si gracias a esta teoría se consigue una red neuronal que pudiera contar con todos los posibles patrones al mismo tiempo?". No es una idea tan alejada de la realidad y esto es debido a la superposición cuántica (propiedad cuántica explicada en el capítulo 2.2.1) que permite tener diferentes estados activos al mismo tiempo. Actualmente esta idea de las redes neuronales cuánticas, documentado por el equipo de investigación de la Universidad de Penn, es solo un marco teórico aún sin implementar. Por otra parte, actualmente, sí es posible simular una sola neurona cuántica en una red neuronal cuántica [8].

En 2018, un equipo de investigación de la Universidad de Pavía en Italia implementó la primera red neuronal de **capa única** del mundo en un ordenador cuántico [8]. En la figura 7.3 se puede observar la red neuronal clásica y en la figura 7.4 se puede observar la red neuronal cuántica en cuestión.

En una red neuronal clásica con una sola neurona, como la mostrada en la figura 7.3, la salida es la suma ponderada de las entradas asignado a la salida binaria a través de una función de activación.

Las redes neuronales cuánticas, como la de la figura 7.4, funcionan de la misma manera, pero la implementación en procesadores cuánticos es diferente. La primera capa de la red cuántica codifica las entradas en un estado cuántico y la segunda capa realiza transformaciones unitarias en la entrada.

[1] La teoría de *Everett* afirma que cada medida "desdobla" nuestro universo en una serie de posibilidades

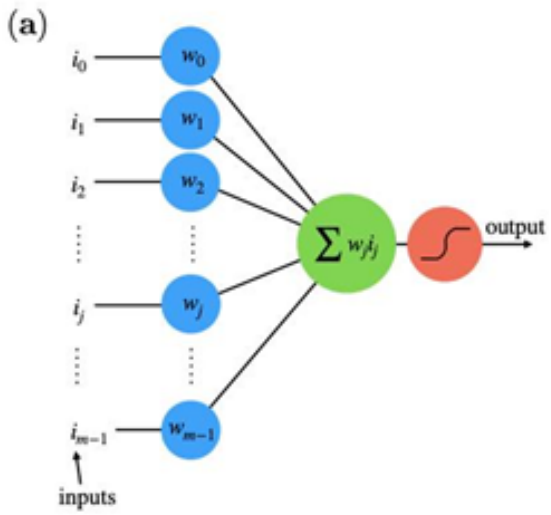


Figura 7.3: Red neuronal clásica. Obtenida de [8].

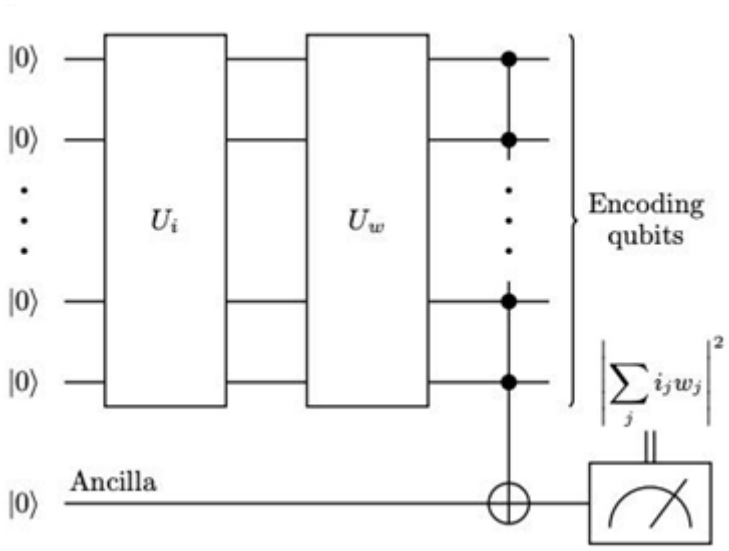


Figura 7.4: Red neuronal cuántica. Obtenida de [8].

7.3.2.1 Ventajas y desventajas de las redes neuronales cuánticas

Según un artículo del equipo de investigación de la Universidad Estatal de Pensilvania, en comparación con las redes neuronales tradicionales, las cuánticas tienen muchas ventajas [8]:

- La cantidad de memoria crece de forma exponencial. Al tratar con cúbits y no con bits la capacidad de la red neuronal aumenta, esto es debido a que los bits pueden almacenar un valor y los cúbits 2. Esto fue explicado en el capítulo 2.

- Se aumenta el rendimiento debido a la menor necesidad de neuronas. Se necesitarían menos neuronas ya que se podrían conseguir los patrones sin la necesidad de operar con tantas neuronas. Esto se podría deber en parte a una consecuencia de la teoría de *Huge Everett* explicada anteriormente en este capítulo.

- El aprendizaje es más rápido gracias a las propiedades de la mecánica cuántica, por lo que los tiempos de espera por el aprendizaje son mucho menores.

- El tamaño es más pequeño gracias al entorno cuántico.

Estas ventajas resuelven prácticamente la mayoría de las limitaciones de las redes neuronales tradicionales. La limitación principal que puede tener una red neuronal es el tiempo invertido en obtener los mejores patrones para el problema en concreto. Si el problema es grande y la red neuronal requiere de mucho tiempo para ser entrenada entonces el tiempo es una limitación. La computación cuántica puede romper esas limitaciones ya que como se ha comentado durante todo el proyecto, la computación cuántica realiza operaciones a mayor velocidad que la computación clásica.

Por otra parte, las redes neuronales cuánticas también tienen desventajas o inconvenientes a remarcar:

- La tecnología cuántica necesaria todavía esta lejos de poder ser utilizada en redes neuronales cuánticas ya que aún esta en proceso de investigación y mejora.

- Actualmente solo se cuenta con marcos teóricos sobre las redes neuronales cuánticas por lo que no se cuenta con una implementación estándar a poder utilizar. Aún se sigue en proceso de investigación.

8 Aplicaciones con computación cuántica

Después de explicar aspectos teóricos de la computación cuántica sobre los diferentes ámbitos de la informática, se van a describir algunos experimentos prácticos con el objetivo de hacer ver la progresión y el estado actual de esta nueva tecnología.

Como comentamos en el capítulo 5, hay librerías disponibles para realizar experimentos con las propiedades de la computación cuántica. Para los experimentos, se utilizará la librería **Qiskit** de **Python** para programar. Es necesario recordar que un resultado de un programa ejecutado sobre circuitería clásica será diferente sobre circuitería cuántica. Para comparar esto, también se utilizará el servidor cuántico de IBM, **IBM Quantum computer**, para ver resultados sobre un ordenador cuántico real.

Desde la sección 8.1 hasta la sección 8.3, el código ha sido extraído de fuentes externas. Se ha de aclarar que el objetivo es únicamente comprender el potencial de la computación cuántica y las posibilidades que esta ofrece y no el de realizar programas en circuitería cuántica.

En estos algoritmos se van a utilizar muchas de las puertas cuánticas que se han explicado en la sección 2.3.

Antes de continuar con los algoritmos cabe destacar que no se van a mencionar los tiempos de procesamiento de los algoritmos ya que no serían significativos debido a todo el proceso necesario para ejecutar un algoritmo cuántico mediante el servidor cuántico de IBM. Este proceso se representa en la figura 8.1 que consiste en una serie de pasos que ralentizan el tiempo real de ejecución del algoritmo sobre la circuitería cuántica por lo que realizar comparaciones entre circuitería cuántica y clásica daría como resultado conclusiones incorrectas.

Por otra parte, sí se va comentar aspectos en relación a los diseños de los circuitos y al comportamiento de los cúbits con las puertas cuánticas. Además, se van a explicar en detalle el resultado de los estados de los cúbits y la comparación de utilizar más o menos cúbits en los mismos problemas.

Por último, aclarar que solo se realizará la comparación de ejecutar el algoritmo en ordenador cuántico (hardware cuántico) y simulador cuántico (hardware clásico) en el algoritmo de *Grover*. Esto es debido a que solo se desea hacer ver como las probabilidades obtenidas en un entorno o en el otro pueden variar como consecuencia de ejecutar el algoritmo contando con un hardware cuántico real y uno clásico. El resto de algoritmos se ejecutarán directamente sobre un ordenador cuántico.

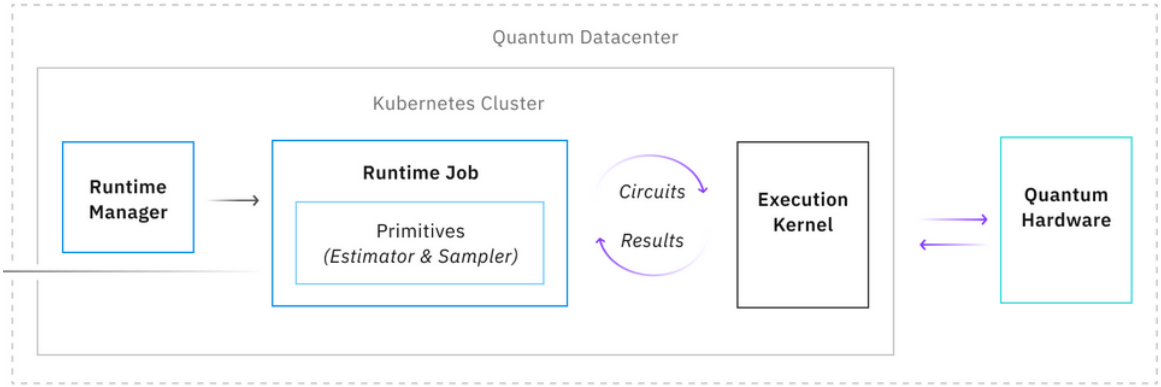


Figura 8.1: Proceso de ejecución de algoritmo cuántico en hardware cuántico de IBM.

8.1 Algoritmo de Grover en Python

Antes de pasar con el algoritmo de *Grover*, cabe recordar, que dicho algoritmo se implementa mediante un "Oráculo" y un "Difusor" que posteriormente se deberán implementar. Este algoritmo se encuentra explicado en la sección 6.2.3 y en la figura 6.3. El código utilizado para el algoritmo de *Grover* se encuentra en [1].

8.1.1 Algoritmo de Grover con 2 cúbits

8.1.1.1 Solución clásica

Inicialmente importamos los módulos de Python para realizar gráficas y optimizar cálculos complejos:

```
1    import matplotlib.pyplot as plt
2    import numpy as np
```

Código 8.1: Modulos necesarios para optimizaciones y gráficos

Acto seguido, se importa el módulo **Qiskit**, el lenguaje de programación desarrollado por IBM para ejecutar código para ordenadores cuánticos. En este módulo se podrán crear circuitos cuánticos, compilaciones de estos y ejecución de dicho código:

```
1    from qiskit import IBMQ, Aer, assemble, transpile, QuantumCircuit, ↩
     ↪ ClassicalRegister, QuantumRegister
2    from qiskit.providers.ibmq import least_busy
3    from qiskit.visualization import plot_histogram
```

Código 8.2: Modulos necesarios para utilizar las funciones necesarias

El siguiente paso es inicializar los cúbits y para ello se colocan en superposición cuántica

utilizando las puertas *Hadamard*. Para ello definiremos una función:

```
# qc: El circuito cuántico con el que se va trabajar
# cubits: El número de cúbits que se van a utilizar

def Inicializar (qc, cubits):
    for q in cubits:
        qc.h(q) # Se aplica la puerta de Hadamard sobre cada cúbit para inicializarlos
    return qc # Se devuelve el circuito cuántico
```

Código 8.3: Función de aplicación de puertas Hadammard sobre los cúbits

A continuación, se comienza con el algoritmo de *Grover* y en este caso lo usaremos con dos cúbits, uno de estado $|0\rangle$ y otro de estado $|1\rangle$, según se observa en la figura 6.3:

```
q = 2 # Dos cúbits
circuito_grover = QuantumCircuit(q) # Creamos circuito cuántico
circuito_grover = Inicializar(circuito_grover,[0,1]) # Inicializamos los cúbits
circuito_grover.draw() # Representación del estado del circuito actual
```

Código 8.4: Ejecución del circuito generado hasta el momento

La ejecución hasta ahora del algoritmo da como resultado el dibujo del circuito que se puede observar en la figura 8.2. En el circuito encontramos los dos cúbits junto con la aplicación de las dos puertas de Hadamard (puerta H).

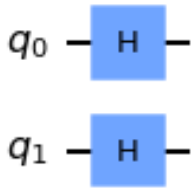


Figura 8.2: Circuito cuántico de *Grover* inicial.

Seguidamente se utiliza el "Oráculo", que en este caso aplica una puerta cuántica Z controlada que agregamos al circuito cuántico:

```
circuito_grover.cz(0,1) # El oráculo se controla mediante la puerta Z-Controlada
```

Código 8.5: Aplicación de la puerta Controlada Z

Por último, se aplica el "Difusor" aplicando las puertas cuánticas H, Z, CZ y H en ambos cúbits.

```
circuito_grover.h([0,1]) # Se aplica la puerta H en los cúbits 0 y 1
circuito_grover.z([0,1]) # Se aplicamos la puerta Z en los cúbits 0 y 1
circuito_grover.cz(0,1) # Se aplicamos la puerta CZ en los cúbits 0 y 1
circuito_grover.h([0,1]) # Se aplicamos la puerta H en los cúbits 0 y 1
```

```
circuito_grover.draw() # Se representa el circuito final
```

Código 8.6: Circuito cuántico de *Grover* final.

Se puede visualizar el circuito final del algoritmo de *Grover* en la figura 8.3.

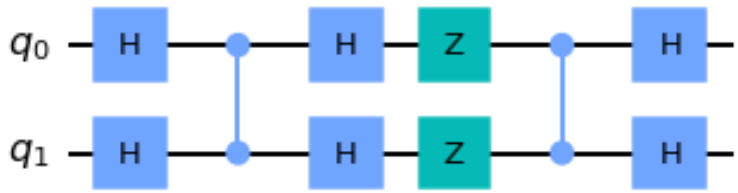


Figura 8.3: Circuito cuántico final de *Grover*.

Se observan los cúbits y se obtienen las probabilidades de estos de ser o 1 ó 0:

```
circuito_grover.measure_all() # Se miden los cúbits del circuito
qasm_simulador = Aer.get_backend('qasm_simulator') #Backend con 'qasm_simulator'
resultados = execute(circuito_grover, backend = qasm_simulador).result() # Ejecutamos el experimento y almacenamos el resultado
respuesta = resultados.get_counts() # Almacenamos las probabilidades de medir cada estado cúbit
plot_histogram(respuesta) # Dibujamos estas probabilidades
```

Código 8.7: Ejecución del algoritmo de *Grover*.

Por último, se muestran las probabilidades, representado en la figura 8.4.

Este experimento se realizó con la expectativa de que el estado sea el $|11\rangle$ midiéndose este con una probabilidad del 100%. Es normal que este valor sea exacto ya que la ejecución del algoritmo se hizo sobre un ordenador clásico. Contar con un simulador cuántico, en este caso, nos proporciona poder interactuar con las propiedades de los cúbits y las puertas cuánticas, peto no un hardware cuántico real. A continuación se presenta el algoritmo ejecutado sobre un ordenador cuántico.

8.1.1.2 Solución cuántica

Una vez realizado el experimento sobre una computadora clásica, se decide realizar su comparación en cuanto a la exactitud de los resultados en una computadora cuántica real, utilizando *IBM Quantum*. Para ello, se accede a la aplicación de *IBM Quantum Lab*, es decir, podemos acceder a los ordenadores cuánticos de *IBM* para realizar experimentos con hardware cuántico [50].

Utilizaremos el mismo circuito ya implementado, pero ahora se monta el escenario necesario para ejecutarlo sobre hardware cuántico. Añadimos el código necesario para autenticarnos en

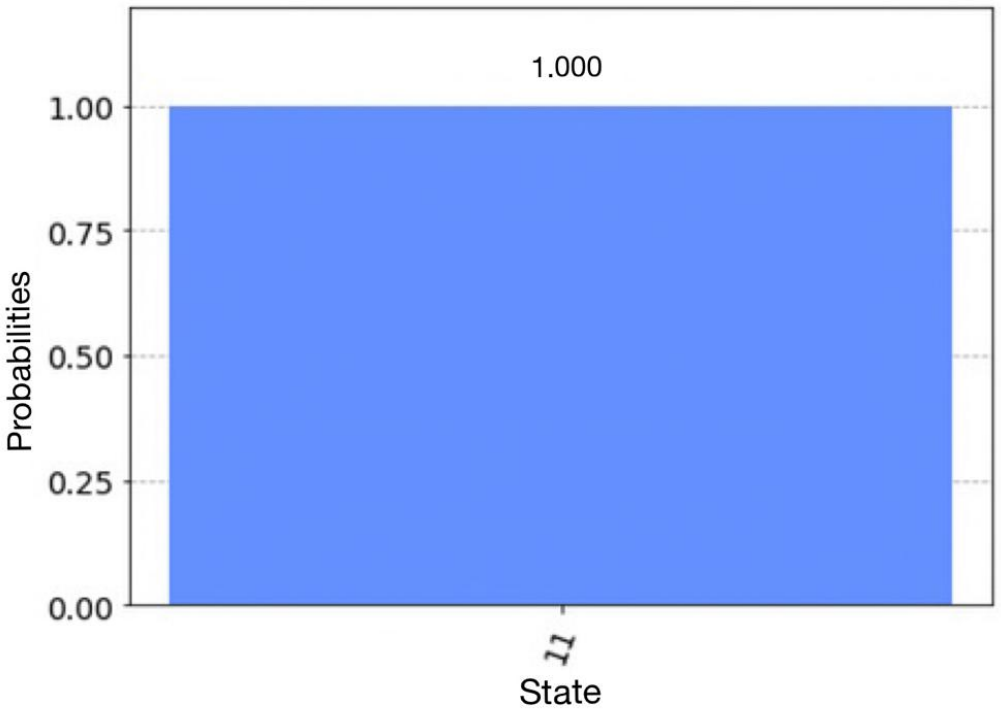


Figura 8.4: Resultado del algoritmo de *Grover* de 2 cúbits con simulador cuántico.

el servidor de IBM y volvemos a cargar el simulador pero ahora específicamente uno cuántico del propio IBM:

```
provider = IBMQ.load_account()
provider = IBMQ.get_provider("ibm-q")
device = least_busy(provider.backends(filters = lambda x: x.configuration().n_qubits >=↩
    ↪  3 and not x.configuration().simulator and x.status().operational==True))
print("Running on current least busy device: ", device)
job = execute(circuito_grover, backend = device, shots = 1024, optimization_level = 3)
from qiskit.tools.monitor import job_monitor
job_monitor(job,interval = 2)
results = job.result()
answer = results.get_counts(circuito_grover)
plot_histogram(answer)
```

Código 8.8: Ejecución del algoritmo de *Grover* sobre el servidor IBM.

Finalmente se obtiene un histograma con las probabilidades obtenidas. Estas probabilidades se pueden observar en la figura 8.5.

Con esto se puede confirmar que en la mayoría de casos, el estado $|11\rangle$ es el considerado. Los otros resultados son debido a errores en la computación cuántica. Estos errores no aparecían en el experimento anterior, en la figura 8.4, ya que se estaba ejecutando sobre hardware clásico. Esto no significa que sea mejor realizarlo sobre ordenador clásico, únicamente estos errores

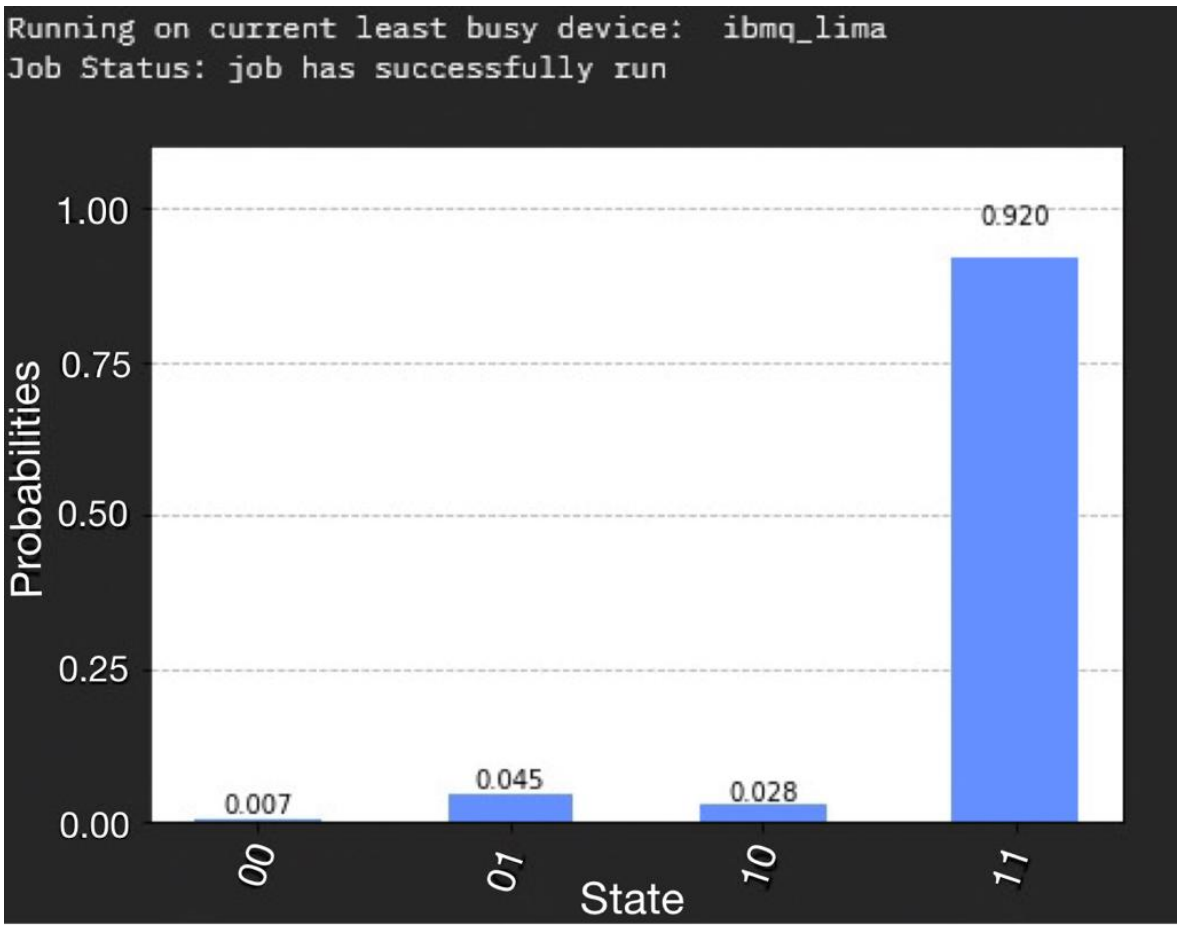


Figura 8.5: Histograma con el resultado del algoritmo de *Grover* de 2 cúbits con ordenador cuántico.

pueden deberse a defectos en los cúbits debido a la decoherencia cuántica (error comentado en el capítulo 2). Como ya se había comentado en el capítulo 2, las mediciones de los cúbits estaban basadas en probabilidades de ser medido como 0 ó ser medido como 1, es por ello que los resultados son también probabilísticos.

8.1.2 Algoritmo de *Grover* con 3 cúbits

Con tres cúbits, los estados que se esperan "ganadores" son $|101\rangle$ y $|110\rangle$, es decir, se esperan dos en vez de uno como pasaba con 2 cúbits. Se ha de recalcar que estos estados esperados son conocidos ya que el circuito se genera con esta disposición de estados ganadores. El circuito a crear es el de la figura 8.6

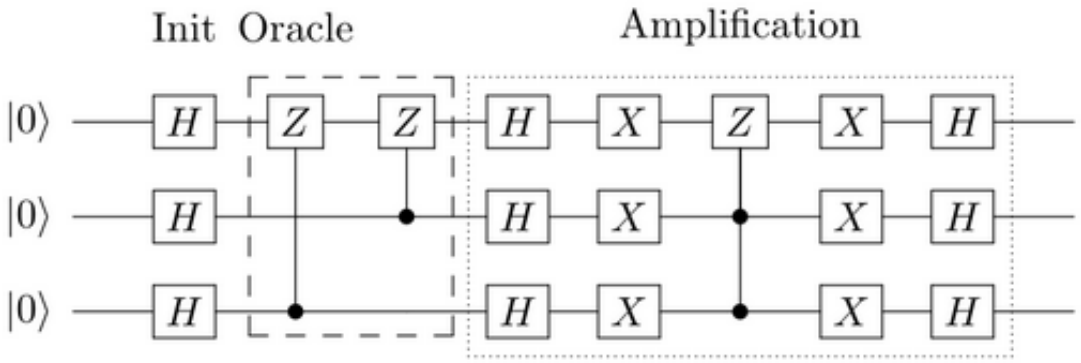


Figura 8.6: Circuito de *Grover* con 3 cúbits. Obtenida en [1].

Como en el caso del algoritmo de *Grover* con 2 cúbits, se importan los módulos necesarios y se define la función de inicialización de cúbits. En este fragmento de código no cambia nada con respecto al algoritmo con 2 cúbits:

```
1 import matplotlib.pyplot as pltl
2 import numpy as ny
3 from qiskit import IBMQ, Aer, execute, transpile, QuantumCircuit, ClassicalRegister, ↩
      ↪ QuantumRegister
4 from qiskit.providers.ibmq import least_busy
5 from qiskit.visualization import plot_histogram
6
7 def Inicializar (qc, cubits):
8     for q in cubits:
9         qc.h(q)
10    return qc
```

Código 8.9: Modulos necesarios y función de aplicación de la puerta de Hadammard

Después se debe aplicar el "Oráculo" específico para 3 cúbits. En este caso, será necesario utilizar dos puertas Z controladas: una que afecte al primer cúbit y al tercer cúbit y otra que afecte al primer cúbit y al segundo cúbit como se representa en la figura 8.6 mediante la etiqueta *Oracle*. De esta forma los 3 cúbits están afectados por la puerta Z controlada.

```
1 qc = QuantumCircuit(3)
2 qc.cz(0,2) # Aplicamos sobre el circuito la puerta CZ en los cúbits 0 y 2
3 qc.cz(1,2) # Aplicamos sobre el circuito la puerta CZ en los cúbits 1 y 2
4 oracle_ex3 = qc.to_gate()
```

```
oracle_ex3.name = "U$_\omega$" # Nombre del oráculo
```

Código 8.10: Aplicamos el Orcáculo en el circuito

A continuación, se define un "Difusor" generalizado para n cúbits:

```
def diffuser(ncubits):
    qc = QuantumCircuit(ncubits)
    for cubit in range(ncubits): # Aplicamos una puerta H a cada cúbit
        qc.h(cubit)

    for cubit in range(ncubits): # Aplicamos una puerta X a cada cúbit
        qc.x(cubit)

    #Aplicamos una puerta MCZ, es decir:
    qc.h(ncubits-1) # Una puerta H al último cúbit
    qc.mct(list(range(ncubits-1)), ncubits -1) # y una puerta MCT para cada cúbit y los↩
    ↪ últimos cúbits
    qc.h(ncubits-1) # y otra puerta H para el último cúbit

    for cubit in range(ncubits): # Aplicamos una puerta X a cada cúbit
        qc.x(cubit)

    for cubit in range(ncubits): # Aplicamos una puerta H a cada cúbit
        qc.h(cubit)
    U_s = qc.to_gate()
    U_s.name = "U$_s$"
    return U_s
```

Código 8.11: Aplicamos el Difusor en el circuito.

Se define el circuito con 3 cúbits y se dibuja:

```
n = 3 # 3 cúbits
circuito_grover = QuantumCircuit(n) # Se crea el circuito con los 3 cúbits
circuito_grover = Inicializar(circuito_grover, [0,1,2]) # Inicializar los cúbits
circuito_grover.append(oracle_ex3, [0,1,2]) # Se junta el circuito inicial con el ↩
    ↪ circuito del oráculo
circuito_grover.append(diffuser(n),[0,1,2]) # Se junta el circuito anterior con el ↩
    ↪ circuito del difusor.
circuito_grover.measure_all()
circuito_grover.draw()
```

Código 8.12: Creamos el circuito con las funciones anteriores creadas

Por tanto, el circuito resultante es el de la figura 8.7, similar al buscado mediante la figura 8.6. Donde se pueden observar los 3 cúbits, seguidamente las 3 puertas *Hadamard*, el oráculo y el difusor. Por último, la medición de los cúbits.

Por último, se simula el experimento con un simulador cuántico y se observa en la figura 8.8 que las probabilidades eran las esperadas.

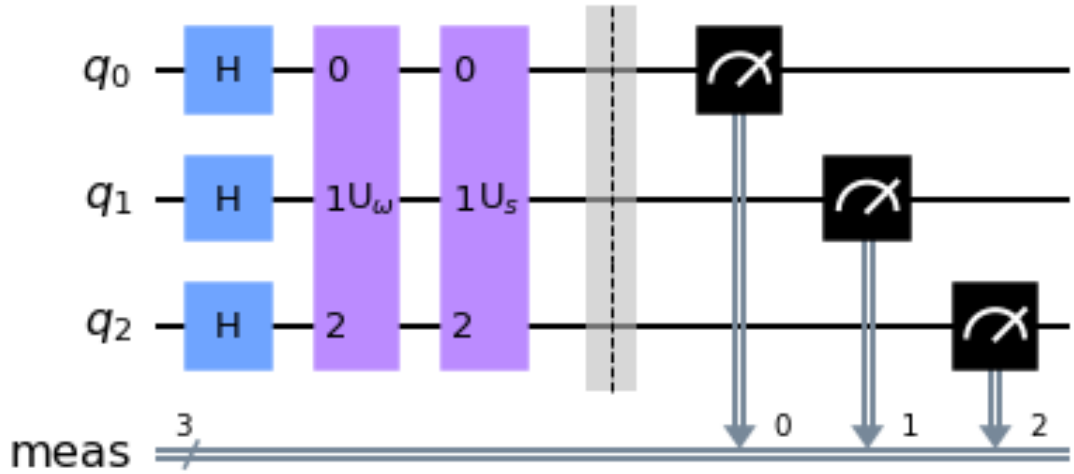


Figura 8.7: Inicializacion, oráculo y difusor.

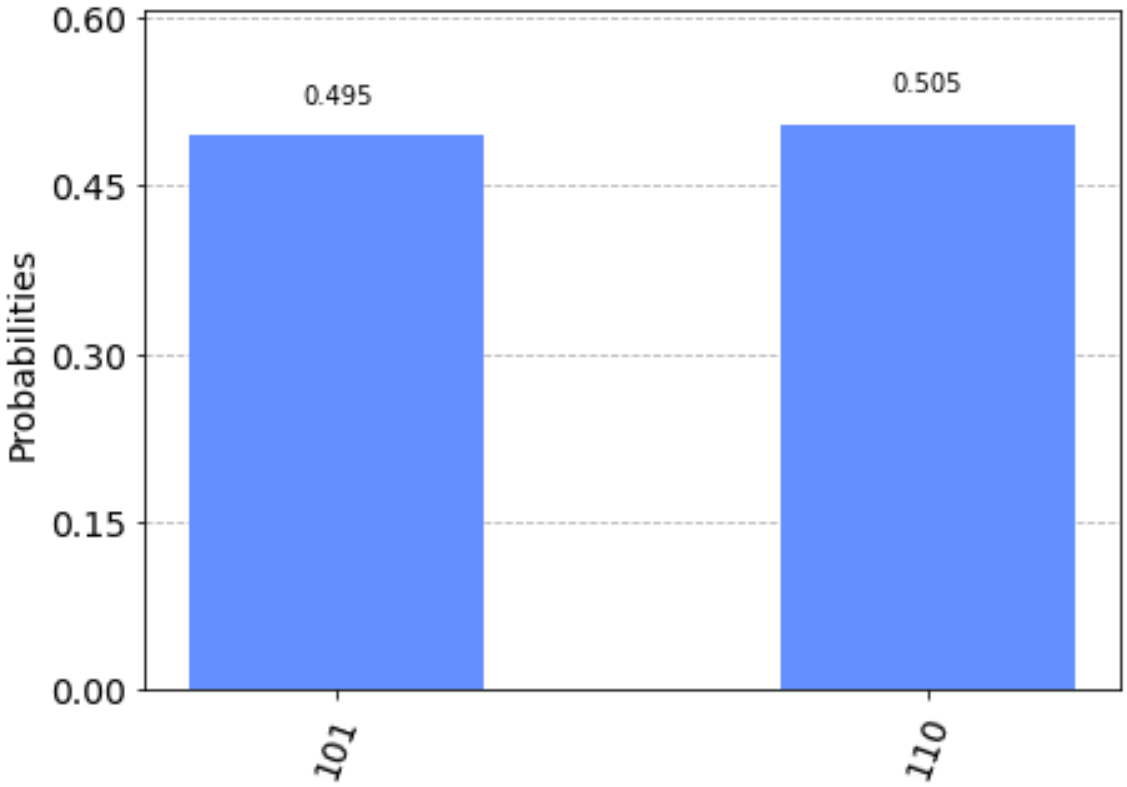


Figura 8.8: Histograma del resultado del algoritmo de *Grover* de 3 cúbits con simulador cuántico.

Ahora se ejecuta el mismo circuito en hardware cuántico, para ello primero se debe montar el escenario cuántico:

```
provider = IBMQ.load_account()
provider = IBMQ.get_provider("ibm-q")
device = least_busy(provider.backends(filters = lambda x: x.configuration().n_cubits >=
    ↪ 3 and not x.configuration().simulator and x.status().operational==True))
print("least busy device: ", device)

from qiskit.tools.monitor import job_monitor
job = execute(circuito_grover, backend = device, shots = 1024, optimization_level = 3)
job_monitor(job,interval = 2)

results = job.result()
answer = result.get_counts(circuito_grover)
plot_histogram(answer)
```

Código 8.13: Ejecutar el circuito en el servidor IBM.

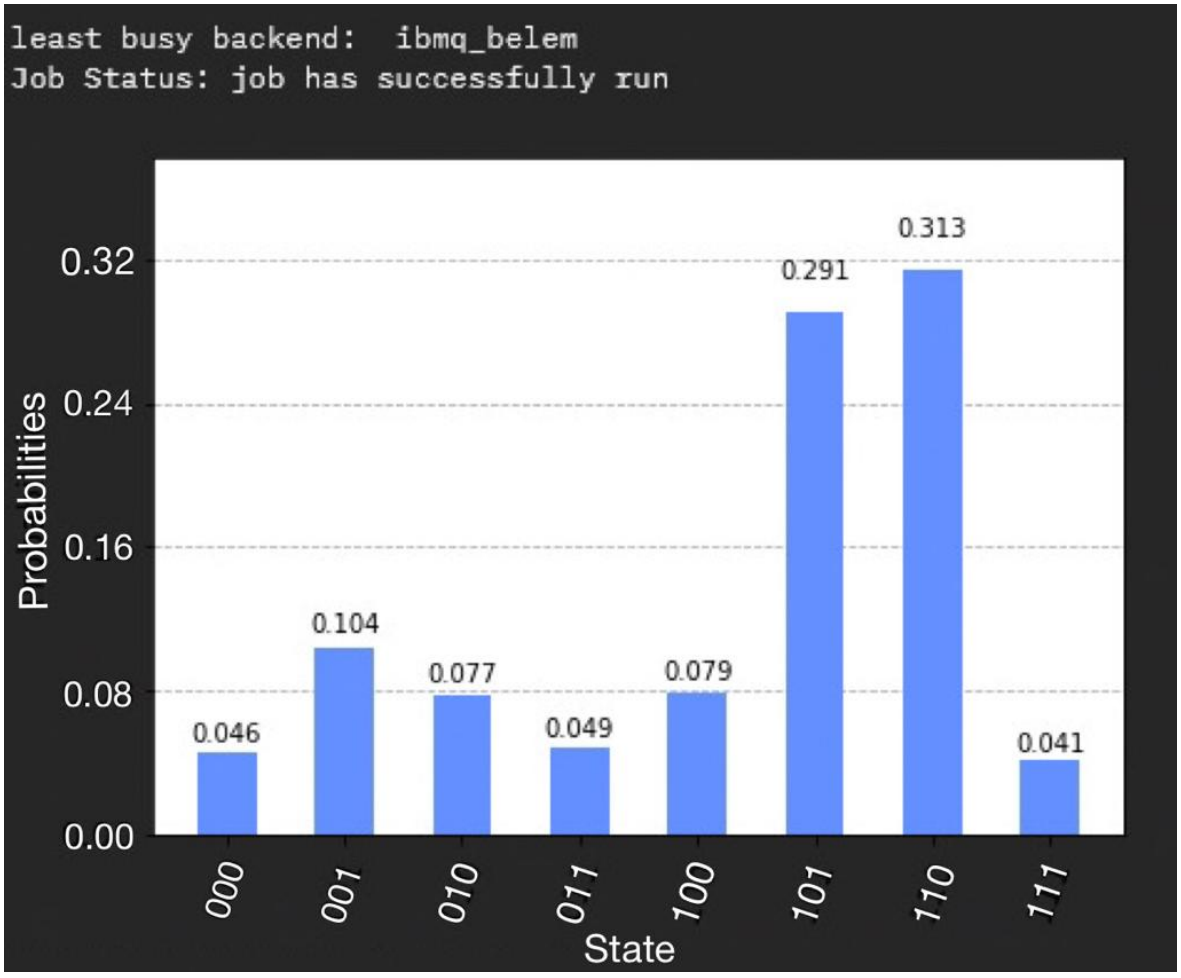


Figura 8.9: Histograma del resultado del algoritmo de *Grover* de 3 cúbits con ordenador cuántico.

Se confirma, mediante la figura 8.9, que cada valor esperado tiene aproximadamente la misma probabilidad de ser medido. Las otras medidas se deben a errores en el cálculo cuántico y probablemente una considerable decoherencia cuántica, al igual que con el experimento con 2 cúbits, ejecutado sobre un ordenador cuántico.

8.2 Algoritmo de Shor en Python

El código utilizado para el algoritmo de *Shor* se encuentra en [2].

Este algoritmo es famoso por poder factorizar números enteros en tiempo polinomial pudiendo así dejar la criptografía RSA inutilizada.

En este ejemplo de implementación del algoritmo de *Shor*, se va a realizar mediante el lenguaje de programación Python con la librería Qiskit, entre otras, para poder simular el circuito cuántico y su ejecución sobre un hardware clásico y cuántico, de un modo similar al anterior caso estudiado.

Se va a implementar únicamente la parte cuántica del algoritmo de *Shor*, es decir, el subprograma para encontrar el periodo que ya se presento en el apartado 6.2.2.2.

Como ya se vio, la función a tratar es la representada en la ecuación 6.8. Además, el circuito que se va a realizar es el representado en la figura 6.2.

En este caso se resolverá el algoritmo de *Shor* para a = 7 y N = 15. Para ello se necesitará de U^x circuitos donde x será las veces que se repetirá el circuito, en este caso 8, la misma cantidad que cúbits que se necesitan pasar por la puerta de *Hadamard* para obtener superposición cuántica (se debe recordar que esta propiedad cuántica fue explicada en la sección 2.2.1) en cada uno de ellos. Se debe definir una función que devuelve dicha puerta U para un número determinado de veces.

Inicialmente importaremos los módulos para utilizar Qiskit:

```
1 import matplotlib.pyplot as pltl
2 import numpy as ny
3 from numpy import pi
4 import pandas as pd
5
6 from qiskit import IBMQ, Aer, transpile, QuantumCircuit, ClassicalRegister, ↩
      ↪ QuantumRegister, assemble
7 from qiskit.providers.ibmq import least_busy
8 from qiskit.visualization import plot_histogram
```

Código 8.14: Módulos necesarios para ejecutar el programa

A continuación, se define la función U^x, este tipo de funciones cuánticas, también llamadas " funciones oráculo", fueron explicadas en la sección 6.2.1.2:

```
def c_amod15(a, power):
    if a not in [2,7,8,11,13]: # el "identificador" de cada cúbit (cúbit 2, cúbit 7,....)
        raise valueError
    U = QuantumCircuit(4)
    for iteration in range(power):
        if a in [2,13]: # el "identificador" de cada cúbit (cúbit 2 y cúbit 13)
            U.swap(0,1)
            U.swap(1,2)
            U.swap(2,3)
        if a in [7,8]: # el "identificador" de cada cúbit (cúbit 7, cúbit 8)
            U.swap(2,3)
            U.swap(1,2)
            U.swap(0,1)
        if a == 11: # el "identificador" del cúbit 11
            U.swap(1,3)
            U.swap(0,2)
        if a in [7,11,13]: # el "identificador" de cada cúbit (cúbit 7, cúbit 11, cúbit 13)
            for q in range(4):
                U,x(q)
    U = U.to_gate()
    U.name = "%i^%i mod 15" % (a, power)
    c_U = U.control()
    return c_U
```

Código 8.15: Definición de función para devolver puerta U necesaria para el algoritmo.

También se debe definir el circuito para QFT, la transformada cuántica de *Fourier*.

```
def qft_dagger(n):
    qc = QuantumCircuit(n)
    for cubit in range(n//2):
        qc.swap(cubit, n-cubit-1)
    for j in range(n):
        for m in range(j):
            qc.cp(pi/float(2**(j-m)), m, j)
        qc.h(j)
    qc.name = "QFT+"
    return qc
```

Código 8.16: Definición de la QFT.

Con estos dos bloques ya se puede proceder a crear el algoritmo de *Shor*:

```
n_count = 8
a = 7
qc = QuantumCircuit(n_count + 4, n_count)

for q in range(n_count):
    qc.h(q)

qc.x(3+n_count)

for q in range(n_count):
```

```
    qc.append(c_amod15(a, 2**q), [q] + [i+n_count for i in range(4)])

qc.append(qft_dagger(n_count), range(n_count))
qc.measure(range(n_count), range(n_count))
qc.draw(fold=-1)
```

Código 8.17: Creación del algoritmo de *Shor*

Seguidamente se muestra el circuito obtenido en la figura 8.10 y se observa como es idéntico al que se estaba buscando, al de la figura 6.2.

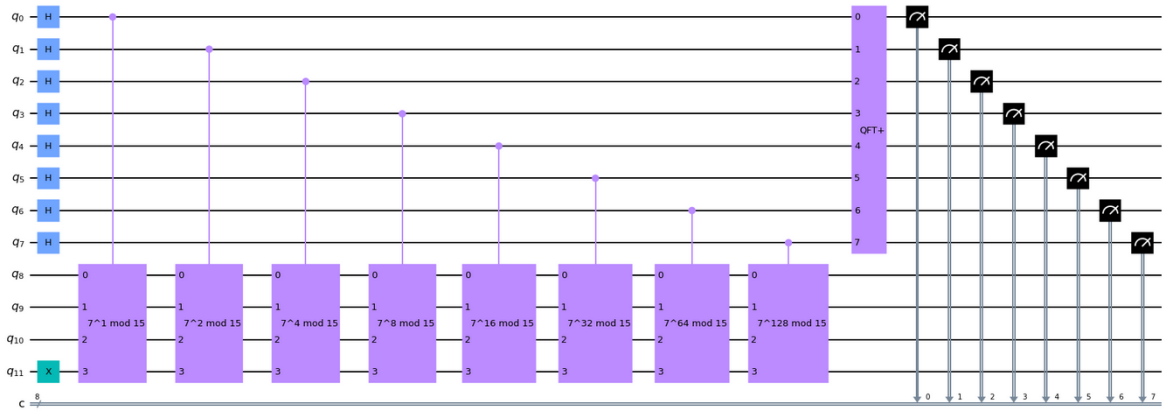


Figura 8.10: Circuito del algoritmo de *Shor*.

A continuación, se realiza la visualización de los resultados del algoritmo. Los resultados pueden observarse en la figura 8.11 y en la figura 8.12.

```
aer_sim = Aer.get_backend('aer_simulator')
t_qc = transpile(qc, aer_sim)
qobj = assemble(t_qc)
results = aer_sim.run(qobj).result()
counts = results.get_counts()
plot_histogram(counts)

rows, measured_phases = [], []
for output in counts:
    decimal = int(output, 2)
    phase = decimal/(2**n_count)
    measured_phases.append(phase)

    rows.append([f"{output}(bin) = {decimal:>3}(dec)", f"{decimal}/{2**n_count} = {↩
    ↪ phase:.2f}"])

headers=["Register Output", "Phase"]
df = pd.DataFrame(rows, columns=headers)
print(df)
```

Código 8.18: Muestra de los resultados del algoritmo

De la figura 8.12 se pueden observar los resultados de las combinaciones de los cúbits

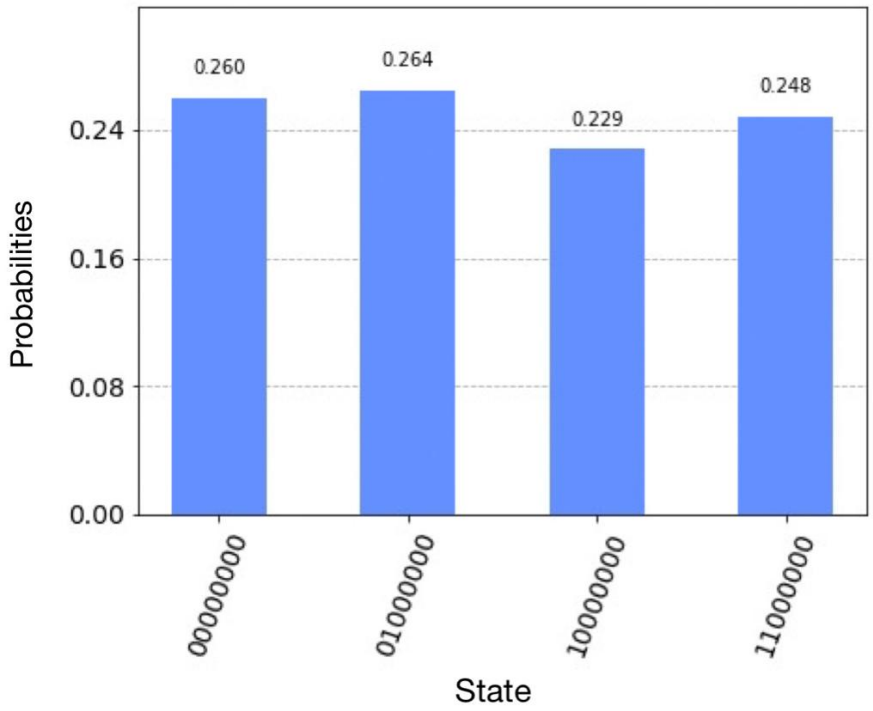


Figura 8.11: Gráfica de los resultados del algoritmo de *Shor*.

	Register Output	Phase
0	00000000(bin) = 0(dec)	0/256 = 0.00
1	10000000(bin) = 128(dec)	128/256 = 0.50
2	11000000(bin) = 192(dec)	192/256 = 0.75
3	01000000(bin) = 64(dec)	64/256 = 0.25

Figura 8.12: Resultados del algoritmo de *Shor*.

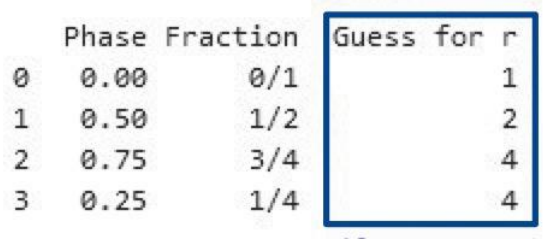
válidos a considerar, siendo "*Register Output*" la combinación de los 8 cúbits en cuestión y la transformación en decimal del número binario de estos. Por último, en "*Phase*" se visualizan los resultados de la división del valor en decimal entre 2^8 siendo 8 el número de los cúbits en superposición cuántica.

Ahora se puede encontrar s y r. Se sabe que r debe ser menor que n por lo que se establece el denominador máximo en 15.

```
rows = []
for phase in measured_phases:
    frac = Fraction(phase).limit_denominator(15)
    rows.append([phase, f"{frac.numerator}/{frac.denominator}", frac.denominator])

headers=["Phase", "Fraction", "Guess for r"]
df = pd.DataFrame(rows, columns=headers)
print(df)
```

Código 8.19: Estimación de r del algoritmo de *Shor*.



	Phase	Fraction	Guess for r
0	0.00	0/1	1
1	0.50	1/2	2
2	0.75	3/4	4
3	0.25	1/4	4

Figura 8.13: Resultados del algoritmo de *Shor* - Estimación para r.

Por último, en la figura 8.13 puede observarse como la "*Fraction*" es equivalente a la "*Phase*". Los denominadores de las fracciones serán los valores buscados para r. Se puede observar, en la figura 8.13 que los valores medios proporcionan $r = 4$. Los otros resultados se pueden deber a que $s = 0$ o porque s y r no son compatibles y en lugar de r obtenemos un factor de r.

8.3 Estimación del número PI mediante el algoritmo de estimación de fase cuántica con Python

El código utilizado para el algoritmo de *Estimación del número PI mediante el algoritmo de estimación de fase cuántica* se encuentra en [61].

Con este algoritmo se intenta obtener el número π lo más aproximado posible a su valor real, para ello se utilizaran diferentes estrategias ya comentadas para diseñar el circuito necesario. Para diseñar el circuito se hará uso de la *Quantum Phase Estimation* (QFE), el cual es un algoritmo cuántico que constituye la base de muchos otros algoritmos cuánticos más complejos. Este experimento se realizará con un número concreto de cúbits y posteriormente se aumentará el número de cúbits utilizados para poder observar como utilizando más cúbits los resultados obtenidos para π son mas exactos.

Inicialmente, se importan los módulos necesarios:

```
from IPython.display import clear_output
from qiskit import *
from qiskit.visualization import plot_histogram
import numpy as np
import matplotlib.pyplot as plotter
from qiskit.tools.monitor import job_monitor

import seaborn as sns, operator
sns.set_style("dark")

pi = np.pi
```

Código 8.20: Módulos necesarios para la ejecución del programa.

Seguidamente se calcula la transformada cuántica de *Fourier* inversa, la cual ya se ha utilizado en otros algoritmos:

```
def qft_dagger(circ_, n_cubits):
    """n-cubit QFTdagger the first n_cubits in circ"""
    for cubit in range(int(n_cubits/2)):
        circ_.swap(cubit, n_cubits-cubit-1)
    for j in range(0,n_cubits):
        for m in range(j):
            circ_.cp(-np.pi/float(2**(j-m)), m, j)
        circ_.h(j)
```

Código 8.21: Definicion de la QFT

A continuación, se especifica la función para preparar el estado inicial para la estimación, la cual ya se ha utilizado en otros algoritmos también:

```
def qpe_pre(circ_, n_cubits):
    circ_.h(range(n_cubits))
    circ_.x(n_cubits)

    for x in reversed(range(n_cubits)):
        for _ in range(2**(n_cubits-1-x)):
            circ_.cp(1, n_cubits-1-x, n_cubits)
```

Código 8.22: Preparación del estado inicial para la estimación

Por comodidad, se define otra función que se encargue de ejecutar el circuito cuántico.

```
def run_job(circ, backend, shots=1000, optimization_level=0):
    t_circ = transpile(circ, backend, optimization_level=optimization_level)
    qobj = assemble(t_circ, shots=shots)
    job = backend.run(qobj)
    job_monitor(job)
    return job.result().get_counts()
```

Código 8.23: Función para ejecutar el circuito

A continuación, se introducen las credenciales IBM para ejecutar los resultados en el hardware cuántico.

```
my_provider = IBMQ.load_account()
simulator_cloud = my_provider.get_backend('ibmq_qasm_simulator')
device = my_provider.get_backend('ibmq_16_melbourne')
simulator = Aer.get_backend('aer_simulator')
```

Código 8.24: Ejecución del circuito en el servidor de IBM

Por último, utilizamos el resto de funciones en una función dedicada a la estimación de PI utilizando *ncubits* de entrada.

```python
def get_pi_estimate(ncubits):

    # create the circuit
    circ = QuantumCircuit(ncubits + 1, ncubits)
    # create the input state
    qpe_pre(circ, ncubits)
    # apply a barrier
    circ.barrier()
    # apply the inverse fourier transform
    qft_dagger(circ, ncubits)
    # apply  a barrier
    circ.barrier()
    # measure all but the last cúbits
    circ.measure(range(ncubits), range(ncubits))

    # run the job and get the results
    counts = run_job(circ, backend=simulator, shots=10000, optimization_level=0)
    # print(counts)

    # get the count that occurred most frequently
    max_counts_result = max(counts, key=counts.get)
    max_counts_result = int(max_counts_result, 2)

    # solve for pi from the measured counts
    theta = max_counts_result/2**ncubits
    return (1./(2*theta))
```

Código 8.25: Función para ejecutar el cometido del programa. Utiliza el resto de funciones definidas anteriormente.

Ahora se ejecuta la función con diferentes cantidades de cúbits:

```python
nqs = list(range(2,12+1))
pi_estimates = []
for nq in nqs:
    thisnq_pi_estimate = get_pi_estimate(nq)
    pi_estimates.append(thisnq_pi_estimate)
    print(f"{nq} cubits, pi  {thisnq_pi_estimate}")
```

Código 8.26: Ejecución de la función anterior para el cálculo de la estimación de PI

Se van obteniendo como resultado los valores de PI cada vez más cercanos, utilizando de 2 a 12 cúbits progresivamente en cada ejecución. Los resultados pueden observarse en la figura 8.14.

Por otra parte, una ejecución de 2 a 15 cúbits da como resultado un valor de PI mucho más exactos. Este resultado puede observarse en la figura 8.15.

Como conclusión, en la figura 8.16, se puede observar cómo cuantos más cúbits se utilizan en la estimación, más exacta es esta. A partir de los 10 cúbits la estimación empieza a ser muy exacta.

```
Job Status: job has successfully run
2 qubits, pi ≈ 2.0
Job Status: job has successfully run
3 qubits, pi ≈ 4.0
Job Status: job has successfully run
4 qubits, pi ≈ 2.6666666666666665
Job Status: job has successfully run
5 qubits, pi ≈ 3.2
Job Status: job has successfully run
6 qubits, pi ≈ 3.2
Job Status: job has successfully run
7 qubits, pi ≈ 3.2
Job Status: job has successfully run
8 qubits, pi ≈ 3.1219512195121952
Job Status: job has successfully run
9 qubits, pi ≈ 3.1604938271604937
Job Status: job has successfully run
10 qubits, pi ≈ 3.1411042944785277
Job Status: job has successfully run
11 qubits, pi ≈ 3.1411042944785277
Job Status: job has successfully run
12 qubits, pi ≈ 3.1411042944785277
```

Figura 8.14: Resultados de estimación de PI con 12 cúbits.

```
Job Status: job has successfully run
2 qubits, pi ≈ 2.0
Job Status: job has successfully run
3 qubits, pi ≈ 4.0
Job Status: job has successfully run
4 qubits, pi ≈ 2.6666666666666665
Job Status: job has successfully run
5 qubits, pi ≈ 3.2
Job Status: job has successfully run
6 qubits, pi ≈ 3.2
Job Status: job has successfully run
7 qubits, pi ≈ 3.2
Job Status: job has successfully run
8 qubits, pi ≈ 3.1219512195121952
Job Status: job has successfully run
9 qubits, pi ≈ 3.1604938271604937
Job Status: job has successfully run
10 qubits, pi ≈ 3.1411042944785277
Job Status: job has successfully run
11 qubits, pi ≈ 3.1411042944785277
Job Status: job has successfully run
12 qubits, pi ≈ 3.1411042944785277
Job Status: job has successfully run
13 qubits, pi ≈ 3.1411042944785277
Job Status: job has successfully run
14 qubits, pi ≈ 3.1411042944785277
Job Status: job has successfully run
15 qubits, pi ≈ 3.1417066155321187
```

Figura 8.15: Resultados de estimación de PI con 15 cúbits.

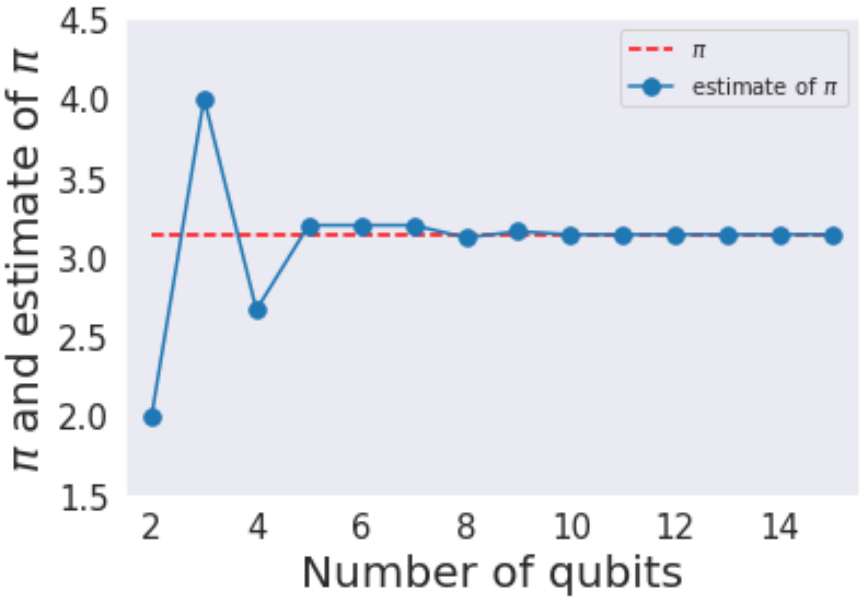


Figura 8.16: Gráfica con la estimación de PI por número de cúbits utilizados.

8.4 Circuitos cuánticos con IBM Composer

En esta sección se van a mostrar dos circuitos, uno aplicando superposición cuántica y otro sin esta. Estos circuitos se han diseñado y utilizado en el servidor de IBM mediante *IBM Composer* y se basan en pequeñas funcionalidades inventadas. El objetivo de estos circuitos es dar a conocer el funcionamiento de las puertas cuánticas junto con los cúbits. Además, también se podrá observar el funcionamiento y el comportamiento de los cúbits junto con las diferentes puertas cuánticas. Cabe destacar que los diseños e implementaciones de los circuitos son un mero ejemplo, se podría perfectamente llegar a las mismas soluciones con otros posibles circuitos. El objetivo principal de esta sección es hacer ver las diferencias de contar con los cúbits en superposición cuántica y los cúbits sin superposición cuántica, de forma que con los cúbits en superposición cuántica se obtendrían diferentes probabilidades en los resultados y sin superposición cuántica se obtendría un único resultado con un 100% de probabilidades.

Es necesario recordar los conceptos explicados en la sección 2.3 sobre las puertas cuánticas y como se representan estas y los cúbits mediante matrices.

8.4.1 Circuito cuántico sin superposición cuántica

En este apartado se pretende analizar un circuito cuántico sin superposición cuántica, es decir, que se obtendrá un único valor con un 100% de probabilidades. El circuito objetivo se representa por la figura 8.17. Por otra parte, el objetivo de este problema es conseguir obtener el valor 10 (en decimal) o el valor 1010 (en binario) utilizando diversas puertas cuánticas con las que los cúbits interactuarán. Al interactuar con las puertas cuánticas, los estados de los cúbits se verán alterados según la puerta cuántica con la que se trabaje. Para este problema, se han utilizado 4 cúbits (q_0, q_1, q_2, q_3).

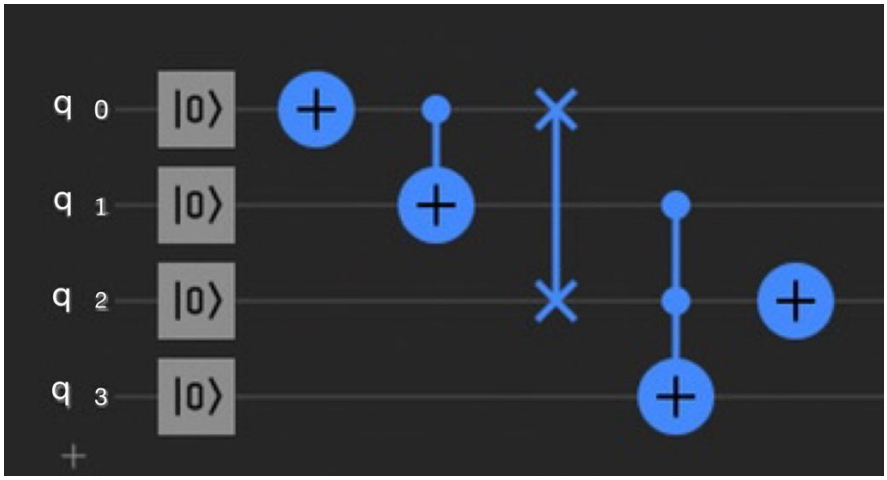


Figura 8.17: Circuto cuántico sin superposición cuántica.

Inicialmente, como se observa en la figura 8.17, todos los cúbits están inicializados a $|0\rangle$. La inicialización a $|0\rangle$ es simplemente por comenzar con algún estado, se podría haber considerado empezar con el estado $|1\rangle$ en los cúbits y haber cambiado el diseño del circuito y se podría llegar al mismo resultado sin problemas. Esta inicialización se representa matemáticamente con la ecuación 8.1.

$$q_0 = q_1 = q_2 = q_3 = |0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{8.1}$$

A continuación, se van aplicando las puertas de forma secuencial a cada cúbit. El primer cúbit, q_0, se le aplica una puerta X (en la imagen 8.17 se representa como el siguiente elemento al "$|0\rangle$" en la fila del cúbit q_0), por lo que su valor se cambiaría por el contrario. La operación con esta puerta cuántica se representa con la ecuación 8.2.

$$X \cdot q_0 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} = |1\rangle, \tag{8.2}$$

siendo X la puerta cuántica X.

Al segundo cúbit, q_1, se le aplica una puerta CNOT (en la imagen 8.17 se representa como el siguiente elemento al "$|0\rangle$" en la fila del cúbit q_1) junto con el cúbit q_0, de forma que q_0 hace de cúbit de control y q_1 es el cúbit que se invertirá según el cúbit de control. Las operaciones necesarias se representan en la ecuación 8.3.

$$\begin{aligned} T = q_0 \otimes q_1 = |1\rangle \otimes |0\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} = |10\rangle \\ CNOT \cdot T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = |11\rangle, \end{aligned} \tag{8.3}$$

siendo CNOT la puerta cuántica CNOT.

Se obtiene el valor de $|11\rangle$ por lo que q_0 tiene el valor de $|1\rangle$ y q_1 tiene el valor de $|1\rangle$.

A continuación, se aplica la puerta SWAP (en la imagen 8.17 se representa como el siguiente elemento al "$|0\rangle$" en la fila del cúbit q_2) sobre los cúbits q_0 y q_2, es decir, se intercambian sus estados. La operación junto con la puerta cuántica SWAP se representa mediante la ecuación 8.4.

$$
\begin{aligned}
T = q_0 \otimes q_2 = |1\rangle \otimes |0\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} = |10\rangle \\
SWAP \cdot T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} = |01\rangle
\end{aligned}
\tag{8.4}
$$

Se obtiene el valor de $|01\rangle$ por lo que q_0 tiene el valor de $|0\rangle$ y q_2 tiene el valor de $|1\rangle$.

Al cuarto cúbit, q_3, se le aplica la puerta *Toffoli* (en la imagen 8.17 se representa como el siguiente elemento al "$|0\rangle$" en la fila del cúbit q_3) con los cúbits de control q_1 y q_2. Las operaciones con la puerta *Toffoli* se representan con la ecuación 8.5.

$$
\begin{aligned}
T = q_1 \otimes q_2 \otimes q_3 = |1\rangle \otimes |1\rangle \otimes |0\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} = |110\rangle \\
TOFFOLI \cdot T = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = |111\rangle
\end{aligned}
\tag{8.5}
$$

Se obtiene el valor de $|111\rangle$ por lo que q_1 tiene el valor de $|1\rangle$, q_2 tiene el valor de $|1\rangle$ y q_3 tiene el valor de $|1\rangle$.

Por último, se aplica la puerta X sobre q_2. La operación se representa mediante la ecuación 8.6.

$$
X \cdot q_2 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} = |0\rangle \tag{8.6}
$$

Una vez finalizadas todas las puertas cuánticas del circuito, se observa en la figura 8.18 como el valor obtenido es el 1010. Este valor es el conjunto de los 4 estados, es decir, q_3 tiene el valor de $|1\rangle$, q_2 tiene el valor de $|0\rangle$, q_1 tiene el valor de $|1\rangle$ y q_0 tiene el valor de $|0\rangle$. En la ecuación 8.7 se puede observar como el conjunto de los estados de los cúbits dan como resultado el valor que se buscaba.

$$q_3, q_2, q_1, q_0 = |1\rangle, |0\rangle, |1\rangle, |0\rangle = 1010_2 = 10_{10} \tag{8.7}$$

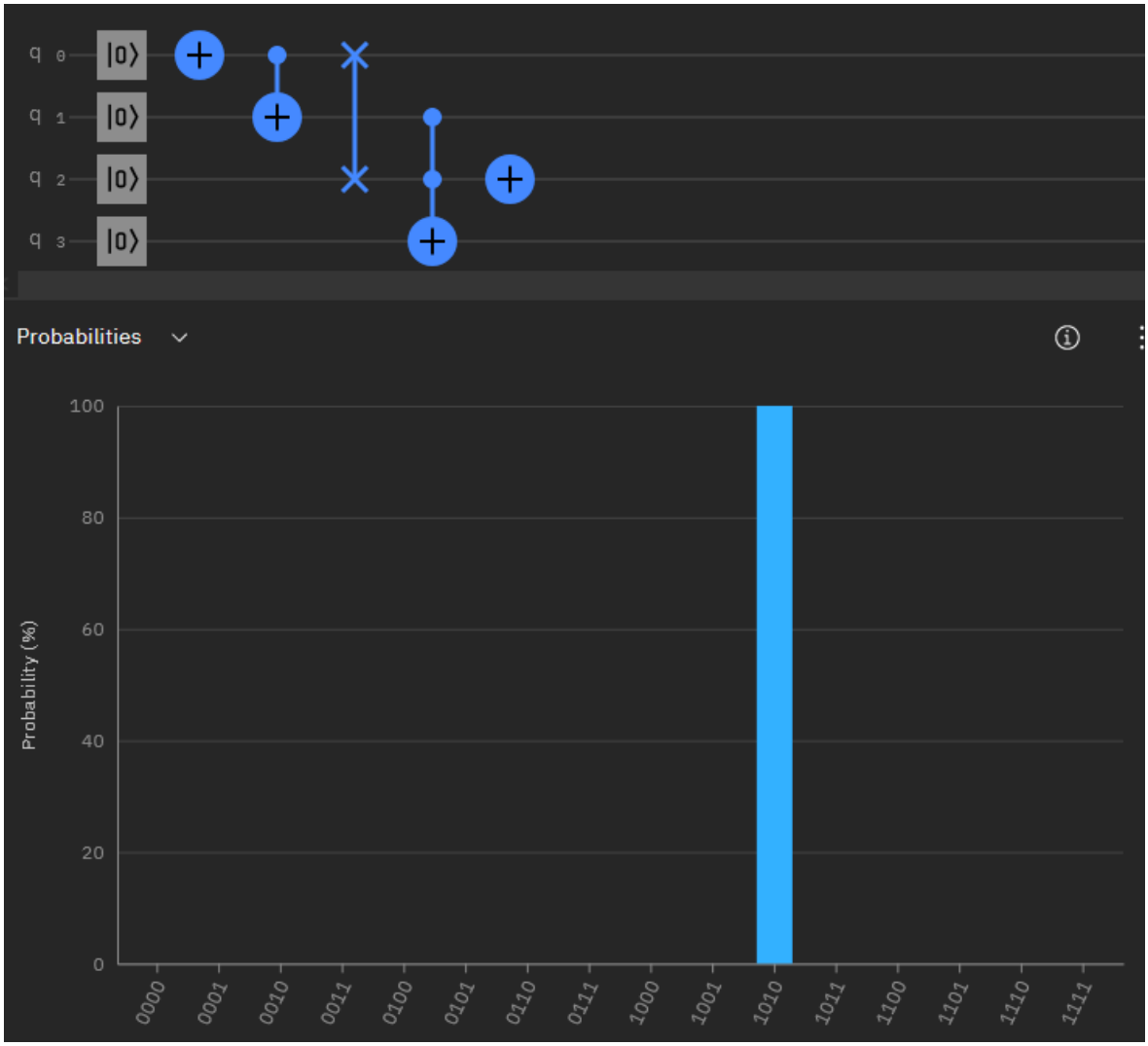


Figura 8.18: Resultado del circuito cuántico sin superposición cuántica.

8.4.2 Circuito cuántico con superposición cuántica

En este apartado se pretende analizar un circuito cuántico con superposición cuántica, es decir, se obtendrán diferentes valores con porcentajes de probabilidades diversos. El circuito a crear puede observarse en la figura 8.19. Con este experimento se quiere hacer ver el funcionamiento de las puertas cuánticas cuando los cúbits están en superposición cuántica. El principal objetivo de este problema es obtener todas las posibles combinaciones en las que el primer cúbit únicamente tenga el valor de 1. A primera vista puede parecer sencillo con unas cuantas puertas cuánticas pero se ha complicado la solución, es decir, se han añadido mas puertas cuánticas de las necesarias para llegar a la misma solución, con el objetivo de demostrar y comprender el funcionamiento de estas y en especial de la superposición cuántica.

La diferencia que hay entre este problema y el anterior es que este sí utiliza superposición cuántica y eso se refleja en el circuito mediante las puertas de *Hadamard*. Estas puertas dan a los cúbits la probabilidad, aproximadamente equitativa, de obtener los valores 0 o 1. En la práctica se consideran las probabilidades de 50% de obtener un 1 y 50% de obtener un 0.

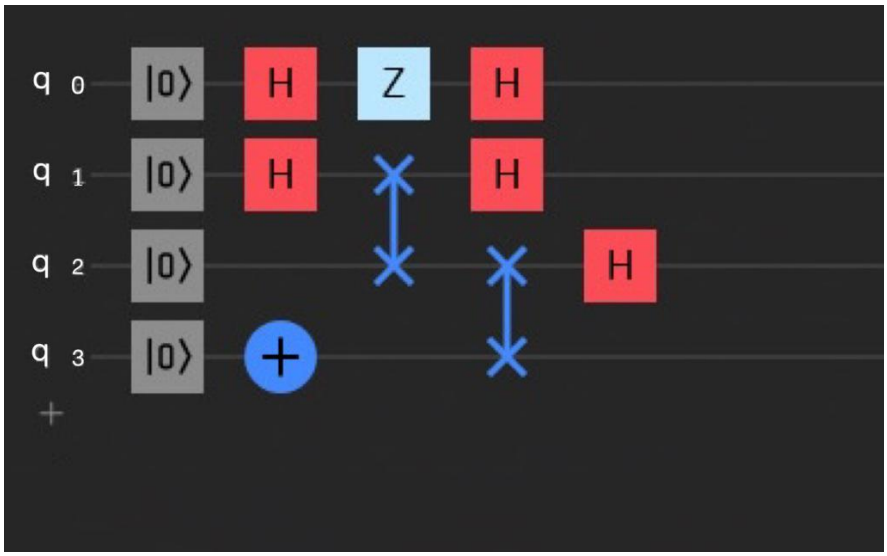


Figura 8.19: Circuito cuántico con superposición cuántica.

En este problema los cálculos no son tan sencillos como en el caso anterior ya que en este se consideran las dos valores por cada cúbit y además al haber puertas cuánticas en paralelo estas se complican bastante. Por ello, en este caso, únicamente se analizarán los cambios y las combinaciones de los valores de los cúbits que se verán a continuación.

Inicialmente, todos los cúbits se les asigna el valor 0, por lo que el estado inicial de los cúbits es el que se representa en la tabla 8.1. Como se ha comentado en el experimento anterior, la inicialización a $|0\rangle$ es simplemente por comenzar con algún estado.

A continuación, se aplica la puerta de *Hadamard* a q_0 y q_1 y la puerta X a q_3. Como se ha comentado anteriormente, la puerta de *Hadamard* hace que dos cúbits puedan tener el

Tabla 8.1: Valores iniciales de los cúbits

cúbits			
$\mathbf{q_3}$	$\mathbf{q_2}$	$\mathbf{q_1}$	$\mathbf{q_0}$
0	0	0	0

valor de 0 y el de 1 pero con diferentes probabilidades. Por otra parte, la puerta X cambia el estado de un cúbit a su opuesto. Los cúbits quedan representados por la tabla 8.2.

Tabla 8.2: Aplicar las puertas *Hadamard* y X sobre los cúbits.

cúbits			
$\mathbf{q_3}$	$\mathbf{q_2}$	$\mathbf{q_1}$	$\mathbf{q_0}$
1 (100%)	0 (100%)	0 (50%)	0 (50%)
-	-	1 (50%)	1 (50%)

En este punto del circuito ya se cuenta con superposición cuánticas sobre los cúbits q_1 y q_0, por lo que las probabilidades de obtener un valor u otro se han multiplicado. Cada valor de la combinación de cúbits tiene un 25% de probabilidad de ser el valor obtenido. Las combinaciones de cada posible resultado serían las reflejadas en la tabla 8.3.

Tabla 8.3: Combinaciones posibles de cúbits junto a sus probabilidades.

cúbits	**Valores**
$q_3q_2q_1q_0$	1000(25%)
	1001(25%)
	1010(25%)
	1011(25%)

A continuación, se añade la puerta Z sobre q_0 y la puerta SWAP sobre q_1 y q_2. La puerta Z realiza un cambio de signo sobre los valores del cúbit pero no afecta a su estado actual. La puerta SWAP realiza el intercambio de los valores de q_1 y q_2. Los nuevos valores son los reflejados en la tabla 8.4. Los posibles valores que se podrían obtener son los observados en la tabla 8.5.

Se vuelve a aplicar las puertas de *Hadamard* en q_0 y q_1 y SWAP entre q_2 y q_3. De forma que los nuevos valores quedan representados por la tabla 8.6.

En este momento, hay que recordar de la sección 2.3 una de las propiedades que tienen las puertas cuánticas y es que todas son **reversibles**, es decir, en q_0 hemos aplicado dos veces la misma puerta *Hadamard* y por tanto hemos obtenido el valor inicial, o en este caso, hemos obtenido el valor dado por la puerta Z y se ha eliminado la propiedad de la superposición cuántica sobre el cúbit q_0. Los valores quedan representados por la tabla 8.7.

Por último, se aplica una última puerta de *Hadamard* para realizar superposición cuántica en q_2. Los valores quedan representados por la tabla 8.8.

Tabla 8.4: Aplicar la puerta Z y la puerta SWAP.

cúbits			
q_3	**q_2**	**q_1**	**q_0**
1 (100%)	0 (50%)	0 (100%)	0 (50%)
-	1 (50%)	-	1 (50%)

Tabla 8.5: Combinaciones posibles de cúbits junto a sus probabilidades.

cúbits	**Valores**
$q_3q_2q_1q_0$	1000(25%) 1001(25%) 1100(25%) 1101(25%)

Con este último cambio se consigue que solo en q_0 siempre sea 1 y el resto vaya variando. Al aplicar 3 cúbits con superposición, aumenta el numero de posibilidades y baja la probabilidad de cada posibilidad, tal y como se observa en la tabla 8.9.

Se obtiene el resultado buscado, como se muestra en la figura 8.20. Todos las opciones del q_0 son 1 y el resto van variando entre todas sus posibilidades.

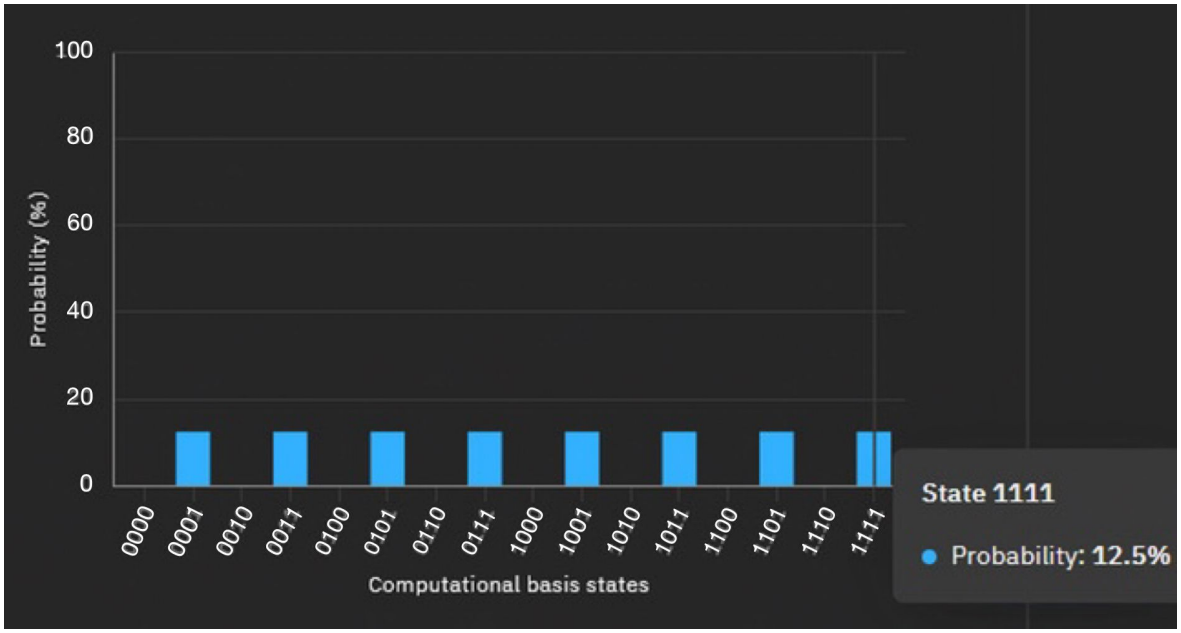


Figura 8.20: Resultado del circuito con superposición cuántica.

Tabla 8.6: Aplicar la puerta Hadammard y SWAP.

cúbits			
$\mathbf{q_3}$	$\mathbf{q_2}$	$\mathbf{q_1}$	$\mathbf{q_0}$
0 (50%)	1 (100%)	0 (50%)	1 (100%)
1 (50%)	-	1 (50%)	-

Tabla 8.7: Combinaciones posibles de cúbits junto a sus probabilidades

cúbits	**Valores**
$q_3q_2q_1q_0$	0101(25%) 0111(25%) 1101(25%) 1111(25%)

Tabla 8.8: Aplicar puerta *Hadamard*.

cúbits			
$\mathbf{q_3}$	$\mathbf{q_2}$	$\mathbf{q_1}$	$\mathbf{q_0}$
0 (50%)	0 (50%)	0 (50%)	1 (100%)
1 (50%)	1 (50%)	1 (50%)	-

Tabla 8.9: Combinaciones posibles de cúbits junto a sus probabilidades.

cúbits	**Valores**
$q_3q_2q_1q_0$	0001 (12.5%) 0011 (12.5%) 0101 (12.5%) 0111 (12.5%) 1001 (12.5%) 1101 (12.5%) 1111 (12.5%)

9 Computación cuántica en la actualidad

La computación cuántica, como ya se ha comentado, es un paradigma que va teniendo peso en la tecnología actual, por lo que aún se están desarrollando nuevos modelos, cambiándolos y mejorándolos, de forma que poco a poco se cuenta con nuevas y mejores versiones de ordenadores cuánticos. A medida que el tiempo pasa, tenemos nuevos ordenadores cuánticos de empresas con las capacidades suficientes para invertir en investigación en este ámbito. A continuación, se van a mencionar algunos de los progresos más recientes en los ordenadores cuánticos en el periodo 2021 - 2022, de esta forma se podrá observar el progreso que estos llevan.

- **Julio de 2021**

  Investigadores de la Universidad de Ciencia y Tecnología de China han desarrollado *Zuchongzhi*, un supercomputador cuántico de 66 cúbits. Según sus desarrolladores, es capaz de realizar una tarea de alta complejidad, que en los supercomputadores actuales más potentes necesitarían de al menos 8 años, en tan solo unas 1,2 horas [39].

- **Noviembre de 2021**

  IBM presenta *Eagle*, su procesador cuántico de 127 cúbits. La potencia de Eagle duplica la de *Zuchongzhi* que hasta el momento era la más avanzada. Este procesador ha superado con creces lo que con procesadores clásicos se era capaz de simular. Según IBM "*el número de bits clásicos necesarios para igualar la potencia de cálculo del procesador de 127 cúbits supera el número total de átomos en los más de 7.500 millones de personas vivas en la actualidad*" [43].

  Además IBM tiene planeado aumentar todavía más esta tecnología y conseguir un nuevo procesador de 433 cúbits para 2022 y uno de 1121 cúbits para 2023 [54].

- **Febrero de 2022**

  Después de *Zuchongzhi* y *Eagle* la supremacía cuántica ha pasado a un segundo plano. El objetivo actual es obtener cúbits de calidad, es decir, cúbits donde se controle o se amenice la decoherencia cuántica que produciría una pérdida de los efectos cuánticos y por tanto todas las ventajas que la cuántica aporta.

  Otro de los objetivos a alcanzar es la corrección de errores ya que la ejecución de algunas operaciones pueden ser incorrectas y por ello es necesario tener un protocolo cuántico de corrección de errores. Grupos de investigadores han conseguido obtener una precisión

superior al 99%, es decir, que los errores que el ordenador cuántico comete se realizan con muy poca frecuencia [75]. Estos experimentos se realizaron con uno o dos cúbits por lo que se necesitaría encontrar la manera de escalar esta tecnología a ordenadores con muchos mas cúbits. Los grupos de investigadores aseguran que con errores inferiores al 1% los protocolos cuánticos de corrección de errores trabajan adecuadamente.

- **Mayo de 2022**

  IBM presenta su nuevo objetivo para 2025, contar con un procesador de más de 4000 cúbits construido con múltiples grupos de procesadores escalados modularmente [92].

  Por otra parte, *Intel* fabricará en 2030 un nuevo procesador capaz de resistir el *hackeo* cuántico de los ordenadores cuánticos [37].

10 Conclusión

Los ordenadores cuánticos están en proceso de investigación, y cada año que pasa el progreso en este campo es algo a tener en consideración. A medida que se van descubriendo nuevos aspectos sobre los ordenadores cuánticos y se van mejorando estos, como por ejemplo incrementando el número de cúbits, se puede observar cómo es necesario realizar cambios para la adaptación a este nuevo paradigma, como ya se esta consiguiendo con la criptografía post-cuántica.

La computación cuántica es una fuente muy poderosa la cual se tendrá al alcance de la mano en un futuro no muy lejano. Esta nos trae muchos beneficios en aspectos de seguridad y sobre todo en aspectos de velocidad de procesamiento, lo cual es el punto crítico que hoy en día estamos teniendo con la computación clásica pero que con la computación cuántica podremos solventar con facilidad.

No hay duda que la computación cuántica marcaría un antes y un después en muchos contextos como el mundo de la inteligencia artificial, e influiría también en otros aspectos como la economía o la medicina, pero referidos a un ámbito tecnológico. La inteligencia artificial será una de las grandes beneficiadas en gran medida todo el potencial de la computación cuántica.

Cabe destacar que la computación cuántica no son todo cosas buenas. Actualmente todas las empresas, organizaciones o entidades utilizan la computación clásica. Lo que migrar, transformar o adaptar la computación actual a la computación cuántica será un problema complejo y un reto a tener en cuenta.

La computación cuántica es una realidad cada vez mucho más cercana y supondría un cambio muy importante en nuestra tecnología y en todos los campos que hacen uso de ella.

10.1 Reflexión personal

Durante toda mi vida he ido observando cómo procesos tan necesarios como la investigación de nuevos fármacos, la investigación de nuevas curas o el desarrollo de modelos de inteligencia artificial, entre otros, conllevan un tiempo de procesamiento para su resolución muy alto, que ralentiza el progreso. Esto puede llegar a ser frustrante, ya que la espera puede ser muy alta para el resultado que podemos obtener y que, quizás, no sea el esperado o no llegue a tiempo para resolver un determinado problema.

Siempre he considerado el "tiempo" un factor muy importante, no solo para la velocidad de procesamiento, sino para los humanos en sí. Muchos algoritmos y estrategias conllevan mucho tiempo, tiempo que quizás una persona no tiene si se dedica a la investigación y se desea obtener grandes avances. O incluso no ya a la investigación, también podemos hablar de empresas que requieren de ejecución de procesos que conllevan mucho tiempo como el aprendizaje de la inteligencia artificial o como la obtención de una vacuna efectiva, entre otras.

A medida que pasa el tiempo, la tecnología avanza y con ella avanza la innovación. Cuanto más avanza la tecnología, más datos se tienen y se almacenan para poder analizarlos. Por lo que, cuantos más datos tenemos no solo sobre personas, sino en general, más nos cuesta llegar a conclusiones y realizar análisis, ya que debemos valorar la máxima cantidad de datos posible. Por lo tanto, la velocidad de procesamiento va siendo menor a medida que la cantidad de datos que tenemos a disposición aumenta llegando a un punto en el que tantos datos se hacen inmanejables por la cantidad de tiempo que la tecnología requiere para obtener las conclusiones que nosotros necesitamos, o para realizar el trabajo que se ha demandado.

Esta evolución de tecnología y cantidad de datos, a día de hoy, se está empezando a notar en sus limitaciones, debido a la velocidad de computación con la que hoy contamos. Es por ello, que la computación cuántica poco a poco se esta viendo que mejorará todos estos aspectos y limitaciones que contamos con la computación clásica.

Personalmente, confío en que la computación cuántica supondrá un cambio en nuestra tecnología pero realmente pienso que no todo el mundo tendrá acceso a los ordenadores cuánticos, es decir, no se podrá tener un ordenador cuántico personal como los ordenadores clásicos que tenemos actualmente en nuestros hogares. Principalmente porque la tecnología cuántica, para funcionar correctamente, debe tener unas condiciones muy particulares, que ya se habían descrito anteriormente, como las temperaturas necesarias para ejecutar un ordenador cuántico o la necesidad de aislar todo el ecosistema para evitar perturbaciones en los cúbits, entre otros. Es por ello, que a menos que se consiga cambiar o adaptar estas condiciones a otros entornos más manejables por cualquier persona en su hogar, los ordenadores cuánticos únicamente estarán disponibles por organizaciones o empresas importantes que lo requieran. De forma que tanto la computación clásica como la computación cuántica puedan coexistir. Esta solución es muy favorable ya que no todo el mundo necesita el potencial del ordenador cuántico, por lo que no todo el mundo tiene porque tener uno. Por otra parte, si que es bastante conveniente que estos se puedan comunicar, de forma que el ordenador cuántico se pueda aprovechar para resolver rápidamente problemas que uno clásico no podría hacer tan fácilmente.

Otro de los inconvenientes que veo en el uso de los ordenadores cuánticos, es su gran dificultad de aprendizaje en estos. Entender cómo funciona es muy diferente a entender cómo funciona un ordenador corriente. Al igual que el estilo de programación es muy diferente, ya que ahora no contamos con bits sino con cúbits, los cuales también tienen propiedades muy diferentes a los bits. Esto quiere decir que lo más probable es que se necesite modificar la forma de programar o por lo menos adaptar el lenguaje de bajo nivel que se comunica con el lenguaje máquina y a partir de ahí generar un lenguaje más abstracto que se pueda

comunicar con una máquina cuántica. Sea como sea, mi impresión es que deberá cambiar, en cierta forma, la manera de ver la informática.

Todos estos avances en la investigación de la computación cuántica los llevan a cabo varios países siendo los pioneros Estados Unidos y China. España esta muy por detrás, ya que su investigación en este ámbito es prácticamente nulo. De hecho, en el año 2021 se anunció que a finales de 2022 se espera tener listo un laboratorio cuántico con ordenadores de 1 y 2 cúbits y para finales de 2025 ordenadores de 20 cúbits. Mientras que Estados Unidos ya piensa en su próximo ordenador cuántico para dentro de un par de años de unos 1100 cúbits. Esto marca la baja inversión en España e incluso en Europa para investigaciones tan necesarias como lo es la computación cuántica.

"*Los que apuesten hoy por el talento y la investigación en computación cuántica serán quienes dentro de diez años puedan cosechar los beneficios y las recompensas que traerá consigo*" [29].

10.1.1 Consideraciones finales

Principalmente, comentar que tras un extenso análisis y recopilación de datos sobre diferentes aspectos de la computación cuántica, se ha conseguido cumplir con todos los objetivos propuestos. Desde, como ya se ha mencionado, la recopilación de los datos, pasando por el análisis de estos y terminando con unas conclusiones que hacen posible poder comprender los beneficios que la computación trae. Este siempre ha sido el objetivo principal y esencial de este proyecto, dar a entender los cambios tan importantes que se obtienen con la computación cuántica. Durante todo el trabajo se han aportado gran cantidad de ejemplos, experimentos e información que hacen posible conseguir todos los objetivos buscados en el trabajo.

Si tuviera que comentar algún aspecto que me hubiera gustado mostrar o realizar en este trabajo habría sido presentar experimentos más sofisticados donde se pudieran observar los grandes beneficios de la computación cuántica. Diseñar y ejecutar experimentos conocidos de complejidad temporal elevada, como por ejemplo, "el problema de la mochila" en ordenadores cuánticos para valorar y comparar los tiempos obtenidos de esta ejecución con una ejecución en un ordenador clásico. Este fue uno de los objetivos principales que me planteé cuando decidí realizar el trabajo sobre la computación cuántica. Diseñar el problema de la mochila y ejecutarlo en un ordenador cuántico y en uno clásico para mostrar los grandes beneficios que podría aportar y además sobre un problema real y actual. Esta motivación inicial dio como consecuencia una intensa investigación en la computación cuántica. Fue así como me di cuenta que actualmente lo que perseguía quizás fuera un objetivo muy ambicioso actualmente ya que no se disponen de las herramientas ni de las tecnologías necesarias hoy en día para conseguir estas metas.

El estado actual de la computación cuántica abre las puertas para investigar más a fondo en los diferentes ámbitos de la informática. A medida que esta tecnología se vaya desarrollando más, la cantidad de posibles investigaciones y experimentos aumentará y además pudiendo obtener resultados mucho más satisfactorios.

A partir de este trabajo se podrían realizar investigaciones más profundas sobre ámbitos concretos. Por ejemplo, realizar una investigación mayor sobre la criptografía cuántica, trabajos específicos sobre la inteligencia artificial en la computación cuántica, análisis exhaustivo sobre los diferentes algoritmos cuánticos, diseño de circuitos cuánticos eficientes, problemas criptográficos en los dispositivos IoT con la llegada de la computación cuántica, entre muchos otros. Como se puede observar, las posibilidades de nuevos trabajos y ampliaciones de este son bastante amplias y a medida que se mejora en esta nueva tecnología la cantidad de trabajos posibles a realizar también aumenta en gran medida.

Espero que este trabajo sea una motivación y un impulso para la realización de muchos otros trabajos que podrían surgir a partir de este. La computación cuántica ya es una realidad, aprovecharnos de su potencial podría ser un salto tecnológico revolucionario para toda la sociedad.

Bibliografía

[1] Algoritmo de grover cuantico. https://qiskit.org/textbook/ch-algorithms/grover.html. [Online; accedido en Diciembre 4, 2021].

[2] Algoritmo de shor cuantico. https://qiskit.org/textbook/ch-algorithms/shor.html. [Online; accedido en Diciembre 4, 2021].

[3] Ciberseguridad cuántica. https://ciberseguridad.com/guias/nuevas-tecnologias/computacion-cuantica/. [Online; accedido en Enero 8, 2022].

[4] Computación cuántica. https://www.dw.com/es/para-qu%C3%A9-sirve-en-realidad-una-computadora-cu%C3%A1ntica/a-50991735. [Online; accedido en Diciembre 28, 2021].

[5] Criptografía cuántica. https://protecciondatos-lopd.com/empresas/criptografia-cuantica/. [Online; accedido en Enero 4, 2022].

[6] Criptogrfia postcuántica. https://futuroelectrico.com/criptografia-poscuantica/. [Online; accedido en Enero 8, 2022].

[7] Criptogrfia postcuántica. https://cmapscloud.ihmc.us/rid=1T0708L5V-9B65SD-6WX/G4%20-.pdf. [Online; accedido en Enero 8, 2022].

[8] El futuro de la inteligencia artificial. https://programmerclick.com/article/3513859777/. [Online; accedido en Diciembre 4, 2021].

[9] El qubit y el algoritmo de shor. http://pimedios.jesussoto.es/2015/04/23/el-qubit-y-el-algoritmo-de-shor/. [Online; accedido en Diciembre 4, 2021].

[10] Inteligencia artificial. https://www.iic.uam.es/inteligencia-artificial/machine-learning-deep-learning/. [Online; accedido en Enero 26, 2022].

[11] Inteligencia artificial cuántica. https://futuroelectrico.com/inteligencia-artificial-y-computacion-cuantica/. [Online; accedido en Enero 24, 2022].

[12] Internet cuántico. https://www.muyinteresante.es/tecnologia/articulo/que-es-el-internet-cuantico-y-que-usos-podria-tener-en-un-futuro-391619818679. [Online; accedido en Enero 10, 2022].

[13] Iot. https://iot.ieee.org/newsletter/Enero-2021/the-convergence-of-iot-and-quantum-computing. [Online; accedido en Enero 26, 2022].

[14] Lenguajes cuánticos. https://lovtechnology.com/lenguajes-de-programacion-para-computacion-cuantica/. [Online; accedido en Enero 21, 2022].

[15] Machine learning qiskit. https://qiskit.org/textbook/ch-machine-learning/machine-learning-qiskit-pytorch.html. [Online; accedido en Diciembre 18, 2021].

[16] Problemas de la computación cuántica. https://www.usmp.edu.pe/publicaciones/boletin/fia/info54/cuantica.html. [Online; accedido en Diciembre 30, 2021].

[17] Problemas de la computación cuántica. https://lacomputtacioncuantica.weebly.com/desventajas-del-computo-cuantico.html. [Online; accedido en Enero 2, 2022].

[18] Qubit. https://www.youtube.com/watch?v=ilPfvMEOmCs. [Online; accedido en Diciembre 28, 2021].

[19] Qubit. https://tecnologia.cibertux.com/2019/11/09/que-es-un-qubit-y-que-tiene-que-ver-con-la-computacion-cuantica/. [Online; accedido en Diciembre 28, 2021].

[20] Qubit. https://revistapesquisa.fapesp.br/es/a-nova-onda-dos-qubits/. [Online; accedido en Diciembre 30, 2021].

[21] Qubit. https://tecno.americaeconomia.com/articulos/el-reino-del-cubit-la-caza-de-la-computacion-cuantica. [Online; accedido en Diciembre 30, 2021].

[22] Superposición cuántica. https://blogs.20minutos.es/ciencia-para-llevar-csic/tag/superposicion-cuantica/. [Online; accedido en Diciembre 27, 2021].

[23] AdslZone. Internet cuántico. https://www.adslzone.net/reportajes/internet/que-es-internet-cuantico/. [Online; accedido en Enero 10, 2022].

[24] Pablo Albarracín. El reino del cubit: a la caza de la computación cuántica. https://www.americaeconomia.com/articulos/el-reino-del-cubit-la-caza-de-la-computacion-cuantica. [Online; accedido en Marzo 18, 2022].

[25] Rodrigo Alonso. Qué es un ordenador cuántico y en qué se diferencia de uno normal. https://hardzone.es/reportajes/que-es/ordenador-cuantico/. [Online; accedido en Noviembre 26, 2021].

[26] Arsys. Ordenador cuántico. https://www.arsys.es/blog/computador-cuantico/. [Online; accedido en Enero 2, 2022].

[27] Simon Baier. https://www.baiersimon.eu/2021/04/16/entangling-three-quantum-network-nodes/. [Online; accedido en Noviembre 27, 2021].

[28] Ahmed Banafa. Computación cuántica e ia: una combinación transformacional. https://www.bbvaopenmind.com/tecnologia/mundo-digital/computacion-cuantica-e-ia/. [Online; accedido en Abril 8, 2022].

[29] BBVA. Computación cuántica. https://www.bbva.com/es/computacion-cuantica-5g-y-blockchain-tecnologias-que-marcaran-la-proxima-decada-en-l
[Online; accedido en Febrero 7, 2022].

[30] bbvaopenmind. Computación cuántica. https://www.bbvaopenmind.com/tecnologia/mundo-digital/computacion-cuantica-y-blockchain-mitos-y-realidades/. [Online; accedido en Diciembre 28, 2021].

[31] bbvaopenmind. Física cuántica. https://www.bbc.com/mundo/noticias-52815076. [Online; accedido en Diciembre 30, 2021].

[32] BBVAOpenmind. Internet cuántico. https://www.bbvaopenmind.com/tecnologia/mundo-digital/el-internet-cuantico-explicado/. [Online; accedido en Enero 10, 2022].

[33] bbvaopenmind. Teletransporte cuántico. https://www.bbvaopenmind.com/ciencia/fisica/el-teletransporte-ya-esta-aqui-pero-no-es-lo-que-esperabamos/. [Online; accedido en Diciembre 29, 2021].

[34] BeeDigital. Ordenador cuántico. https://www.beedigital.es/tendencias-digitales/que-es-un-ordenador-cuantico-y-cuales-son-sus-usos/. [Online; accedido en Enero 2, 2022].

[35] bit2me. Ataque del 51 https://academy.bit2me.com/ataque-51-bitcoin/. [Online; accedido en Mayo 10, 2022].

[36] Davide Castelvecchi. Internet cuántico. https://www.nature.com/articles/d41586-021-00420-5. [Online; accedido en Enero 10, 2022].

[37] Borja Colomer. Intel desea sacar una cpu resistente al hackeo cuántico para 2030. https://elchapuzasinformatico.com/2022/05/intel-cpu-resistente-hackeo-cuantico/. [Online; accedido en Mayo 20, 2022].

[38] ComputerHoy. que es criptografía cuántica. https://www.youtube.com/watch?v=UICMBSu5AeA. [Online; accedido en Mayo 8, 2022].

[39] Digitaleye. Zuchongzhi. https://www.digitaleye.uma.es/digital-eye-observatory/zuchongzhi-el-supercomputador-cuantico-capaz-de-realizar-calc
[Online; accedido en Febrero 4, 2022].

[40] Albert Einstein, Boris Podolsky, and Nathan Rosen. Can quantum-mechanical description of physical reality be considered complete? *Physical review*, 47(10):777, 1935.

[41] euclides. Algoritmo de euclides. https://euclides.org/algoritmo-de-euclides/. [Online; accedido en Mayo 14, 2022].

[42] EuropaPress. Primera teleportación sostenida a larga distancia de qubits de fotones. https://www.europapress.es/ciencia/laboratorio/noticia-primera-teleportacion-sostenida-larga-distancia-qubits-fotones-20201229175629.html. [Online; accedido en Mayo 12, 2022].

[43] Made for minds. Ibm crea el ordenador cuántico superconductor más potente de la historia. https://www.dw.com/es/ibm-crea-el-ordenador-cu%C3%A1ntico-superconductor-m%C3%A1s-potente-de-la-historia/a-59837328. [Online; accedido en Abril 9, 2022].

[44] Elena Gil. Big data, privacidad y protección de datos. https://www.aepd.es/sites/default/files/2019-10/big-data.pdf. [Online; accedido en Mayo 15, 2022].

[45] Beatriz Guillén. Así es el superordenador que descifra el genoma humano. https://elpais.com/tecnologia/2017/03/30/actualidad/1490875903_152817.html. [Online; accedido en Abril 9, 2022].

[46] hmong. Algoritmo de grover. https://hmong.es/wiki/Grover%27s_algorithm. [Online; accedido en Mayoo 15, 2021].

[47] hmong. Transformada cuántica de fourier. https://hmong.es/wiki/Quantum_Fourier_transform. [Online; accedido en Mayo 15, 2022].

[48] Hrvwiki. Puertas cuánticas. https://es.hrvwiki.net/wiki/quantum_logic_gate, 2018. [Online; accedido en Noviembre 17, 2021].

[49] IBM. Computador cuántico. https://quantum-computing.ibm.com/composer/files/new. [Online; accedido en Enero 27, 2022].

[50] ibm. Ibm quantum lab. https://quantum-computing.ibm.com. [Online; accedido en Mayo 15, 2022].

[51] Interferencias. Interferencias. https://edwicarval.wixsite.com/fisicaondasyelectro/interferencia-destructiva-y-constructiva. [Online; accedido en Noviembre 24, 2021].

[52] Guillermo Julián. Computación cuántica: qué es, de dónde viene y qué ha conseguido. https://www.xataka.com/ordenadores/computacion-cuantica-que-es-de-donde-viene-y-que-ha-conseguido. [Online; accedido en Abril 9, 2022].

[53] Steve Jurvetson. Un ordenador cuántico romperá el cifrado rsa de 2048 bits en ocho horas. https://www.technologyreview.es/s/11209/un-ordenador-cuantico-rompera-el-cifrado-rsa-de-2048-bits-en-ocho-horas. [Online; accedido en Mayo 8, 2022].

[54] Juan Carlos López. En la carrera por el mejor ordenador cuántico hay dos bandos, y no son estados unidos y china: son las dos tecnologías de cúbits más avanzadas. https://www.xataka.com/investigacion/carrera-mejor-ordenador-cuantico-hay-dos-bandos-no-estados-unidos-china-dos-tecnolog [Online; accedido en Mayo 20, 2022].

[55] Microsoft. Clasificador cuántico. https://docs.microsoft.com/es-es/azure/quantum/user-guide/libraries/machine-learning/intro. [Online; accedido en Diciembre 4, 2021].

[56] Microsoft. Qubit. https://azure.microsoft.com/es-es/overview/what-is-a-qubit/#introduction. [Online; accedido en Diciembre 27, 2021].

[57] notinforma. Silq es un nuevo lenguaje de programación para computadoras cuánticas – techcrunch. https://notinforma.com/silq-es-un-nuevo-lenguaje-de-programacion-para-computadoras-cuanticas-techcrunch/. [Online; accedido en Mayo 13, 2022].

[58] notinforma. Tesis de church-turing. https://hmong.es/wiki/Church_Turing_Thesis. [Online; accedido en Mayo 13, 2022].

[59] Javier Pastor. Ley de moore. https://www.xataka.com/robotica-e-ia/antes-teniamos-ley-moore-ahora-tenemos-ley-huang-que-perfila-futuro-nvidia-arm. [Online; accedido en Diciembre 30, 2021].

[60] Qiskit. https://qiskit.org/textbook/ch-algorithms/deutsch-jozsa.html#2.-Worked-Example--. [Online; accedido en Noviembre 30, 2021].

[61] Qiskit. Estimating pi using quantum phase estimation algorithm. https://qiskit.org/textbook/ch-demos/piday-code.html. [Online; accedido en Mayo 15, 2022].

[62] Qiskit. Redes neuronales híbridas cuánticas-clásicas con pytorch y qiskit. https://qiskit.org/textbook/ch-machine-learning/machine-learning-qiskit-pytorch.html. [Online; accedido en Mayo 15, 2022].

[63] Sacyr. Ordenador cuántico. https://www.sacyr.com/-/asi-se-construye-un-computador-cuantico. [Online; accedido en Enero 2, 2022].

[64] Seas. Ordenador cuántico. https://www.seas.es/blog/informatica/que-es-un-ordenador-cuantico-y-su-funcionamiento/. [Online; accedido en Diciembre 27, 2021].

[65] SINC. Machine learning. https://www.agenciasinc.es/Noticias/Hasta-donde-llega-el-aprendizaje-automatico-cuantico. [Online; accedido en Diciembre 1, 2021].

[66] SoloEsCiencia. https://soloesciencia.com/2018/02/27/entrelazamiento-amor-al-mas-puro-estilo-cuantico/. [Online; accedido en Noviembre 27, 2021].

[67] Speqtral. Fibre-qkd. https://speqtral.space/fibre-qkd/. [Online; accedido en Mayo 10, 2022].

[68] Jaime Señor Sánchez. Aplicacion de sistemas post- cuanticos a la seguridad en nodos de internet of things. https://oa.upm.es/66693/1/TFM_JAIME_SENOR_SANCHEZ.pdf. [Online; accedido en Mayoo 15, 2021].

[69] Cibertux Advanced Technology. Esfera de bloch. https://tecnologia.cibertux.com/tag/qubits/. [Online; accedido en Noviembre 22, 2021].

[70] Thales. Ciberseguridad cuántica. https://www.thalesgroup.com/es/alemania/magazine/computacion-cuantica-ciberseguridad. [Online; accedido en Enero 4, 2022].

[71] Carme Torras. Turing: el nacimiento del hombre (1912), la máquina (1936) y el test (1950). https://blogs.elpais.com/turing/2012/07/turing-el-nacimiento-del-hombre-1912-la-maquina-1936-y-el-test-1950.html, 7 2012. [Online; accedido en Marzo 7, 2022].

[72] @FlashTweet Twitter. Ordenador cuántico zuchongzhi. https://twitter.com/search?src=hash&q=%23Zuchongzhi&lang=fr, 2021. [Online; accedido en Noviembre 15, 2021].

[73] Unir. Seguridad informática. https://ecuador.unir.net/actualidad-unir/que-es-seguridad-informatica/. [Online; accedido en Enero 4, 2022].

[74] upwiki. Red neuronal cuántica. https://es.upwiki.one/wiki/Quantum_neural_network. [Online; accedido en Mayo 15, 2022].

[75] Sandra Viñas. Un estudio demuestra que es posible la computación cuántica casi libre de errores: hallan la forma de que exista un 99 https://www.businessinsider.es/computacion-cuantica-99-precision-posible-998395. [Online; accedido en Mayo 6, 2022].

[76] Wikipedia. Aalgoritmo criptográfico. https://es.wikipedia.org/wiki/Algoritmo_criptogr%C3%A1fico. [Online; accedido en Enero 4, 2022].

[77] Wikipedia. Algoritmo cuántico. https://es.wikipedia.org/wiki/Algoritmo_cu%C3%A1ntico. [Online; accedido en Enero 21, 2022].

[78] Wikipedia. Algoritmo de shor. https://es.wikipedia.org/wiki/Algoritmo_de_Shor. [Online; accedido en Enero 8, 2022].

[79] Wikipedia. Algoritmo deutsch. https://es.wikipedia.org/wiki/Algoritmo_de_Deutsch-Jozsa. [Online; accedido en Enero 24, 2022].

[80] Wikipedia. Algoritmo grover. https://bo.wikiqube.net/wiki/Grover%27s_algorithm. [Online; accedido en Enero 24, 2022].

[81] Wikipedia. Algoritmo shor. https://en.wikipedia.org/wiki/Shor%27s_algorithm. [Online; accedido en Enero 24, 2022].

[82] Wikipedia. Criptografía cuántica. https://www.thalesgroup.com/es/alemania/magazine/computacion-cuantica-ciberseguridad. [Online; accedido en Enero 4, 2022].

[83] Wikipedia. Decoherencia cuántica. https://es.wikipedia.org/wiki/Decoherencia_cu%C3%A1ntica. [Online; accedido en Enero 2, 2022].

[84] Wikipedia. Inteligencia artificial. https://es.wikipedia.org/wiki/Aprendizaje_autom%C3%A1tico_cu%C3%A1ntico. [Online; accedido en Enero 26, 2022].

[85] Wikipedia. Inteligencia artificial. https://es.wikipedia.org/wiki/Red_neuronal_cu%C3%A1ntica. [Online; accedido en Enero 26, 2022].

[86] Wikipedia. Inteligencia artificial cuántica. https://pr.wikiqube.net/wiki/Quantum_neural_network. [Online; accedido en Enero 26, 2022].

[87] Wikipedia. Puerta cuántica. https://es.wikipedia.org/wiki/Puerta_cu%C3%A1ntica. [Online; accedido en Enero 21, 2022].

[88] Wikipedia. Árbol de merkel. https://es.wikipedia.org/wiki/%C3%81rbol_de_Merkle#/media/Archivo:Hash_Tree.svg. [Online; accedido en Mayo 10, 2022].

[89] the free encyclopedia Wikipedia. Algoritmo de grover. https://en.wikipedia.org/wiki/Grover%27s_algorithm. [Online; accedido en Diciembre 1, 2021].

[90] the free encyclopedia Wikipedia. Máquina de turing. https://es.wikipedia.org/wiki/M%C3%A1quina_de_Turing, 5 2021. [Online; accedido en Noviembre 11, 2021].

[91] the free encyclopedia Wikipedia. Puertas lógicas. https://es.wikipedia.org/wiki/Puerta_l%C3%B3gica, 10 2021. [Online; accedido en Noviembre 20, 2021].

[92] Noelia Hernández y Alberto Iglesias Fraga. Así será el primer computador cuántico de 4.000 cúbits que cambiará la investigación científica y burlará el cifrado. https://www.elespanol.com/invertia/disruptores-innovadores/innovadores/tecnologicas/20220510/computador-cuantico-cambiara-investigacion-cientifica-burlara-cifrado/671432868_0.html. [Online; accedido en Mayo 20, 2022].

[93] Youtube. https://www.youtube.com/watch?v=zkEwbRzQk6w. [Online; accedido en Noviembre 29, 2021].

[94] Youtube. Algoritmo. https://es.wikipedia.org/wiki/Algoritmo. [Online; accedido en Enero 21, 2022].

[95] Youtube. Puerta cuántica. https://www.youtube.com/watch?v=FNV0F6hdHuI. [Online; accedido en Enero 21, 2022].

[96] Álvaro Rodrigo Reyes Rosado. Estado de la criptografía post-cuántica y simulaciones de algoritmos post-cuánticos. http://openaccess.uoc.edu/webapps/o2/bitstream/10609/89026/6/alvaroreyesTFM1218memoria.pdf#%5B%7B%22num%22%3A63%2C%22gen%22%3A0%7D%2C%7B%22name%22%3A%22XYZ%22%7D%2C82%2C771%2C0%5D. [Online; accedido en Mayo 10, 2022].

[97] Óscar F. Civieta. Los ordenadores cuánticos amenazan el futuro de las criptomonedas: ¿será el fin de la 'blockchain'? https://www.businessinsider.es/acabaran-ordenadores-cuanticos-criptomonedas-966393. [Online; accedido en Mayo 10, 2022].

Lista de Acrónimos y Abreviaturas

BQP	*Bounded-probability Quantum Polynomial.*
CNAG	Centro Nacional de Análisis Genómico.
IBM	*International Business Machines.*
IoT	*Internet of Things.*
QFE	*Quantum Phase Estimation.*
QFT	*Quantum Fourier Transform.*
QKD	*Quantum Key Distribution.*
QNN	*Quantum Neural Network.*