PRINCIPIOS FUNDAMENTALES DE COMPUTACIÓN CUÁNTICA

VICENTE MORET BONILLO

Profesor Titular de Universidad. Senior Member, IEEE.

Departamento de Computación. Facultad de Informática.

UNIVERSIDAD DE A CORUÑA

2013

PRINCIPIOS FUNDAMENTALES DE COMPUTACIÓN CUÁNTICA

ÍNDICE DE CONTENIDOS

--..--

INTRODUCCIÓN

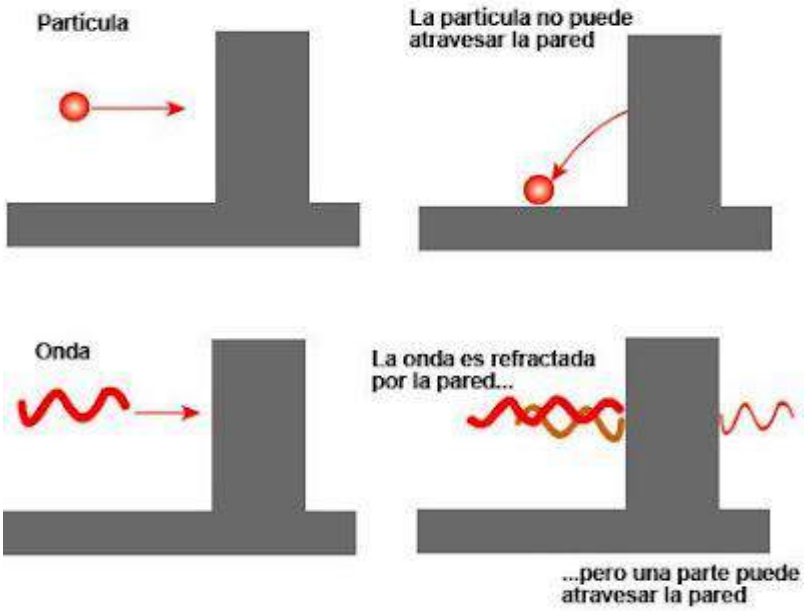
Hacia el inicio de la década de los 60, Rolf Landauer comenzó a preguntarse si las leyes físicas imponían algunas limitaciones al proceso de cómputo. En concreto se interesó sobre el origen del calor disipado por los ordenadores y se preguntó si este calor era algo inherente a las leyes de la física o se debía a la falta de eficiencia de la tecnología disponible.

Rolf W. Landauer (4 de febrero de 1927 - 28 de abril de 1999). "En toda operación lógicamente irreversible que manipula información, como la reinicialización de memoria, hay aumento de entropía, y una cantidad asociada de energía es disipada como calor". Este principio es relevante en informática reversible y en informática cuántica.

El tema parece realmente interesante si recordamos que uno de los problemas de los actuales ordenadores de alta velocidad es la eliminación del calor producido durante su funcionamiento. Por otra parte, a medida que evoluciona la tecnología aumenta la escala de integración y caben más transistores en el mismo espacio.

Cada vez se fabrican microchips más pequeños ya que, cuanto más pequeño es el dispositivo, mayor velocidad de proceso se alcanza. Sin embargo no podemos hacer los chips infinitamente pequeños. Hay un límite en el cual dejan de funcionar correctamente. Cuando se llega a la escala de nanómetros los electrones se escapan de los canales por donde deben circular por el llamado "efecto túnel", un fenómeno típicamente cuántico. Así, y dicho de forma un tanto grosera, si una partícula clásica se encuentra con un obstáculo, lo normal es que no pueda atravesarlo y rebote. Pero los electrones son partículas cuánticas y presentan

comportamiento ondulatorio; por ello, existe la posibilidad de que una parte de tales electrones pueda atravesar las paredes entre las que están confinados. De esta manera la señal puede pasar por canales donde no debería circular y el chip deja de funcionar correctamente.



Efecto túnel. La parte superior de la figura representa la situación descrita por la física clásica. La parte inferior de la figura representa la situación que describe la física cuántica.

En este contexto la computación digital tradicional no debe estar muy lejos de su límite, puesto que ya se ha llegado a escalas de sólo algunas decenas de nanómetros. Estas reflexiones iban a ser el germen de las actuales ideas acerca de la computación cuántica y acerca de los ordenadores cuánticos.

La Evolución de la Computación Cuántica

Las ideas esenciales de la computación cuántica surgieron en los primeros años de la década de 1980 de la mente de Paul Benioff que trabajaba con

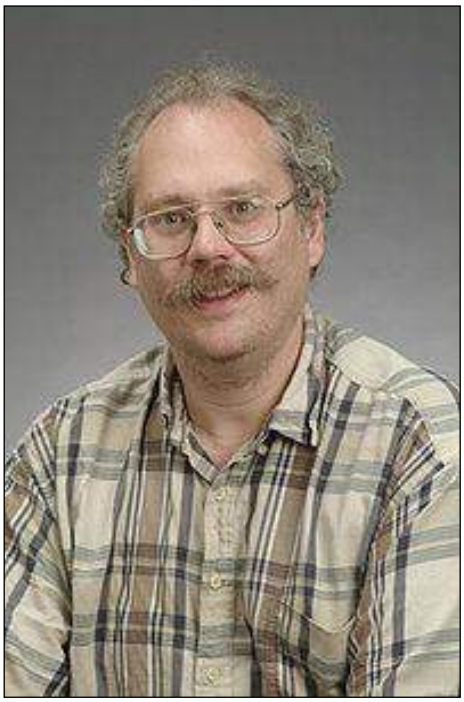
ordenadores tradicionales (máquinas de Turing) a los que hacía operar con algunos de los principios fundamentales de la mecánica cuántica. Entre 1981 y 1982 Richard Feynman proponía el uso de fenómenos cuánticos para realizar cálculos computacionales y exponía que, dada su naturaleza, algunos cálculos de gran complejidad se realizarían más rápidamente en un ordenador cuántico. En 1985 David Deutsch describió el primer computador cuántico universal, capaz de simular cualquier otro computador cuántico (principio de Church-Turing ampliado). De este modo surgió la idea de que un computador cuántico podría ejecutar diferentes algoritmos cuánticos.

Richard Feynman nació el 11 de mayo de 1918 en Nueva York, y murió el 15 de febrero de 1985. Feynman fue un influyente popularizador de la física a través de sus libros y conferencias, y un ejemplo más de ello fue la charla que dio en 1959 sobre nanotecnología, titulada Hay mucho lugar al fondo. Feynman ofreció 1.000 dólares en premios por dos de sus retos en nanotecnología. También fue uno de los primeros científicos en señalar las posibilidades de los ordenadores cuánticos. Muchas de sus clases luego se convirtieron en libros, como El carácter de la ley física y Electrodinámica cuántica: La extraña teoría de la luz y la materia. Entre sus trabajos más importantes, destaca la elaboración de los diagramas de Feynman, una forma intuitiva de visualizar las interacciones de partículas atómicas en electrodinámica cuántica mediante aproximaciones gráficas en el tiempo. Feynman es considerado también una de las figuras pioneras de la nanotecnología, y una de las primeras personas en proponer la realización futura de las computadoras cuánticas. Pero tal vez el homenaje más relevante no proviene de los premios académicos: poco después de su muerte, un grupo de estudiantes de Caltech escaló el frente de la Biblioteca Millikan de la universidad y colgó un gran cartel de tela con la leyenda "We love you Dick!"

A lo largo de los años 90 la teoría empezó a plasmarse en la práctica, y aparecen los primeros algoritmos cuánticos, las primeras aplicaciones cuánticas y las primeras máquinas capaces de realizar cálculos cuánticos. En 1993 Dan Simon demostraba la ventaja que tendría un computador cuántico frente a uno tradicional al comparar el modelo de probabilidad clásica con el modelo cuántico. Sus ideas sirvieron como base para el desarrollo de algunos algoritmos de

auténtico interés práctico, como el de Shor. También en 1993, Charles Benett descubre el tele-transporte cuántico, que abre una nueva vía de investigación hacia el desarrollo de comunicaciones cuánticas.

Entre 1994 y 1995 Peter Shor definió el algoritmo que lleva su nombre y que permite calcular los factores primos de números a una velocidad mucho mayor que en cualquier computador tradicional. Además su algoritmo permitiría romper muchos de los sistemas de criptografía utilizados actualmente. Su algoritmo sirvió para demostrar a una gran parte de la comunidad científica, que observaba incrédula las posibilidades de la computación cuántica, que se trataba de un campo de investigación con un gran potencial. Además, un año más tarde, propuso un sistema de corrección de errores en el cálculo cuántico.



Peter Shor Williston (nacido el 14 de agosto, 1959) es un profesor estadounidense de matemáticas aplicadas en el MIT, famoso por su trabajo en computación cuántica, en particular por elaborar el algoritmo de Shor, un algoritmo cuántico de factorización exponencialmente más rápido que el mejor algoritmo conocido actualmente que se ejecuta en un ordenador clásico.

En 1996 Lov Grover propone el algoritmo de búsqueda de datos que lleva su nombre. Aunque la aceleración conseguida no es tan drástica como en los cálculos factoriales o en simulaciones físicas, su rango de aplicaciones es mucho

mayor. Al igual que el resto de algoritmos cuánticos, se trata de un algoritmo probabilístico con un alto índice de acierto.

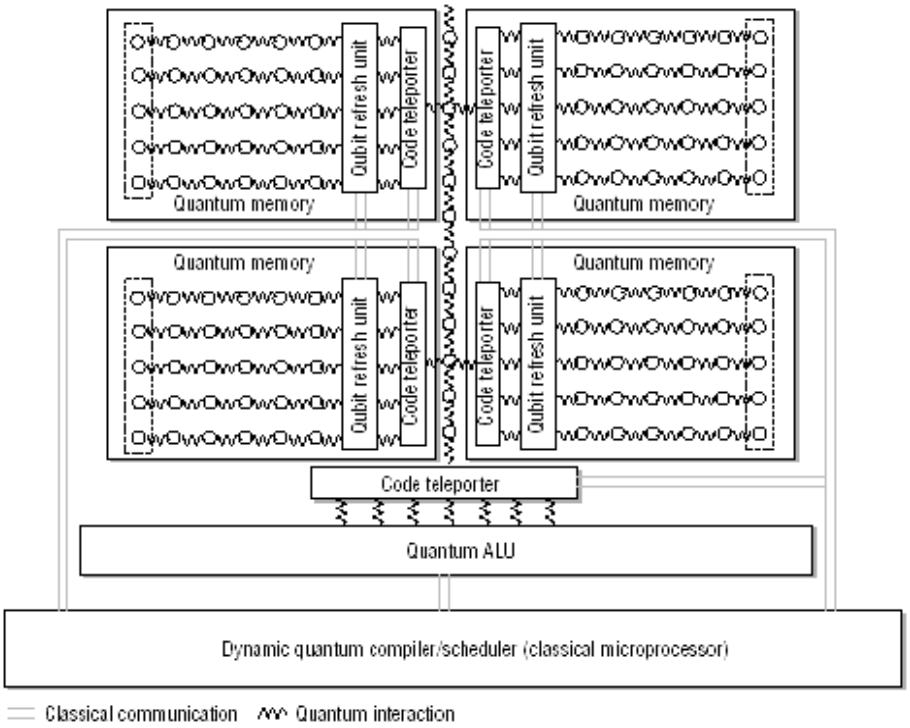
En 1997 se iniciaron los primeros experimentos prácticos y se abrieron las puertas para empezar a implementar todos aquellos cálculos y experimentos que habían sido descritos teóricamente hasta entonces. El primer experimento de comunicación segura usando criptografía cuántica se realiza con éxito a una distancia de 23 Km. Además se realiza el primer tele-transporte cuántico de un fotón. A partir de entonces la computación cuántica es una realidad imparable, y entre 1998 y 1999, investigadores de Los Álamos y del MIT consiguen propagar el primer "qubit" (del inglés 'bit cuántico') a través de una disolución de aminoácidos. Este experimento supuso el primer paso para analizar la información que transporta un qubit.

Una de las propiedades más importantes de la criptografía cuántica es que si un tercero intenta hacer eavesdropping durante la creación de la clave secreta, el proceso se altera advirtiéndose al intruso antes de que se trasmita información privada. Esto es una consecuencia del principio de incertidumbre de Heisenberg, que nos dice que el proceso de medir en un sistema cuántico perturba dicho sistema.

En 1998 nació la primera máquina de 2-qubits, que fue presentada en la Universidad de Berkeley, California. Un año más tarde, en 1999, en los laboratorios de IBM se diseñó la primera máquina de 3-qubits, que además fue capaz de ejecutar por primera vez el algoritmo de búsqueda de Grover.

En el año 2000, de nuevo en IBM, se diseña un computador cuántico de 5-qubits capaz de ejecutar un algoritmo de búsqueda de orden que forma parte del algoritmo de Shor. Este algoritmo se ejecutaba en un simple paso cuando en un

computador tradicional requería numerosas iteraciones. Ese mismo año, científicos de Los Álamos anunciaron el desarrollo de un computador cuántico de 7-qubits.



Representación esquemática de un eventual ordenador cuántico.

En 2001, IBM y la Universidad de Stanford, consiguen ejecutar por primera vez el algoritmo de Shor en el primer computador cuántico de 7-qubits desarrollado en Los Álamos. En el experimento se calcularon los factores primos de 15, dando el resultado correcto de 3 y 5 utilizando para ello 1018 moléculas, cada una de ellas con 7 átomos.

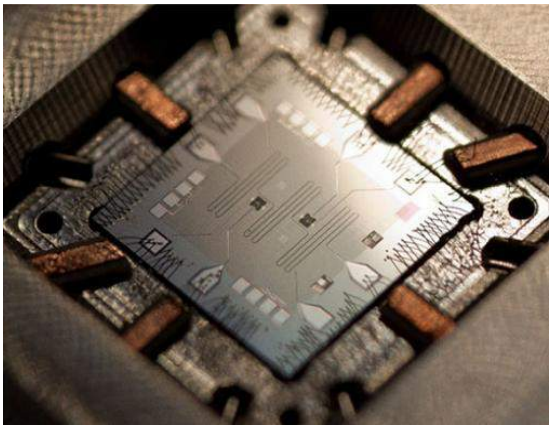
En 2005 el Instituto de "Quantum Optics and Quantum Information" en la Universidad de Innsbruck (Austria) anunció que sus científicos habían conseguido implementar el primer qubyte, una serie de 8 qubits, utilizando trampas de iones.

Y en 2006 científicos en Waterloo y MIT consiguen desarrollar un sistema de 12-qubits.

En septiembre de 2007, dos equipos de investigación estadounidenses, el National Institute of Standards (NIST) de Boulder y la Universidad de Yale en New Haven, consiguieron unir componentes cuánticos a través de superconductores. De este modo aparece el primer bus cuántico, que además puede ser utilizado como memoria cuántica reteniendo la información cuántica durante un corto espacio de tiempo antes de ser transferida a otro dispositivo.

Según la Fundación Nacional de Ciencias (NSF) de los EEUU, en 2008 un equipo de científicos consiguió almacenar por primera vez un qubit en el interior del núcleo de un átomo de fósforo, y pudieron hacer que la información permaneciera intacta durante 1,75 segundos. Este lapso de tiempo puede ser ampliable mediante métodos de corrección de errores, por lo que es un gran avance en el almacenamiento de información.

En 2009 el equipo de investigadores estadounidense dirigido por Robert Schoelkopf, de la Universidad de Yale, crea el primer procesador cuántico de estado sólido, mecanismo que se asemeja y funciona de forma similar a un microprocesador convencional, aunque con la capacidad de realizar sólo unas pocas tareas muy simples, como operaciones aritméticas o búsquedas de datos. La comunicación en el dispositivo se realiza mediante fotones que se desplazan sobre el bus cuántico.



Recreación de un posible procesador cuántico.

Finalmente, en 2011, la primera computadora cuántica comercial es vendida por la empresa D-Wave System a Lockheed Martin, por 10 millones de dólares, e IBM anuncia que ha creado un chip lo suficientemente estable como para permitir que la informática cuántica llegue en breve plazo a hogares y empresas. Se estima que en unos 10 ó 12 años se puedan estar comercializando los primeros sistemas cuánticos "asequibles".

Cuestiones Preliminares

La computación cuántica es un paradigma de computación distinto al de la computación clásica. Se basa en el uso de qubits en lugar de bits, y da lugar a nuevas puertas lógicas que hacen posibles nuevos algoritmos. Una misma tarea puede tener diferente complejidad en computación clásica y en computación cuántica, lo que ha dado lugar a una gran expectación, ya que algunos problemas intratables pasan a ser tratables. Mientras un computador clásico equivale a una máquina de Turing, un computador cuántico equivale a una máquina de Turing cuántica.

En la computación digital un bit sólo puede tomar dos valores: 0 ó 1. En cambio, en la computación cuántica, intervienen las leyes de la mecánica cuántica, y la partícula, el qubit, puede estar en superposición coherente: puede ser 0, puede ser 1 y puede ser 0 y 1 a la vez (dos estados ortogonales de una partícula subatómica). Eso permite que se puedan realizar varias operaciones simultáneamente, según el número de qubits.

El número de qubits indica la cantidad de bits que pueden estar en superposición. Con los bits convencionales si teníamos un registro de tres bits había ocho valores posibles, y el registro sólo podía tomar uno de esos valores. En cambio, si tenemos un vector de tres qubits, la partícula puede tomar ocho valores distintos a la vez gracias a la superposición cuántica. Así, un vector de tres qubits permitiría un total de ocho operaciones paralelas. Como cabe esperar, el número de operaciones es exponencial con respecto al número de qubits.

Para hacerse una idea del gran avance que esto supone, un computador cuántico de 30 qubits equivaldría a un procesador convencional de 10 teraflops (millones de millones de operaciones en coma flotante por segundo), cuando actualmente las computadoras trabajan en el orden de gigaflops (miles de millones de operaciones en coma flotante por segundo).

Desde una perspectiva práctica, y asumiendo un planteamiento "Informático" de la cuestión, los asuntos que nos interesan de la computación cuántica pueden resumirse en los siguientes puntos: Soporte Físico, Transmisión de Datos, Algoritmos Cuánticos y Arquitecturas y Modelos.

En relación con el primer punto, no se ha resuelto completamente todavía el problema de cuál sería el soporte físico ideal para la computación cuántica que, no obstante, debe cumplir los siguientes requisitos:

1. El sistema ha de poder inicializarse, esto es, llevarse a un estado de partida conocido y controlado.

2. Ha de ser posible hacer manipulaciones a los qubits de forma controlada, con un conjunto de operaciones que forme un conjunto universal de puertas lógicas (para poder reproducir cualquier otra puerta lógica posible).

3. El sistema ha de mantener su coherencia cuántica a lo largo del experimento.

4. Ha de poder leerse el estado final del sistema tras el cálculo.

5. El sistema ha de ser escalable y tiene que haber una forma definida de aumentar el número de qubits para poder tratar con problemas de mayor coste computacional.

En cuanto a la transmisión de datos, científicos de los laboratorios Max Planck y Niels Bohr obtuvieron, no hace mucho, resultados sobre la transmisión de información cuántica usando la luz como vehículo, a distancias de 100 km. Los resultados dan niveles de éxito en las transmisiones del 70%, lo que representa un nivel de calidad que permite utilizar protocolos de transmisión con autocorrección. Actualmente se trabaja en el diseño de repetidores, que permitirían transmitir información a distancias mayores a las ya alcanzadas.

Por otra parte, los algoritmos cuánticos diseñados se basan en un margen de error conocido en las operaciones de base y trabajan reduciendo el margen de error a niveles exponencialmente pequeños, comparables al nivel de error de las máquinas actuales. Algunos ejemplos importantes son el algoritmo de Shor, el algoritmo de Grover, o el algoritmo de Deutsch-Jozsa.

Por último existen varios modelos y arquitecturas dignas de mención, como son la computadora cuántica de Benioff, la computadora cuántica de Feynman, o la computadora cuántica de Deutsch.

Indudablemente la computación cuántica es un campo de interés y gran futuro. Al respecto, se ha sugerido el uso de la computación cuántica como alternativa superior a la computación clásica para varios problemas, entre ellos la factorización de números enteros, problemas de logaritmo discreto, o la simulación de sistemas cuánticos. El siempre genial Richard Feynman conjeturó en 1982 que los ordenadores cuánticos serían eficaces como simuladores universales de sistemas cuánticos, y en 1996 se demostró que la conjetura era correcta. Sin embargo, como en toda disciplina emergente, todavía quedan muchos problemas pendientes de solución.

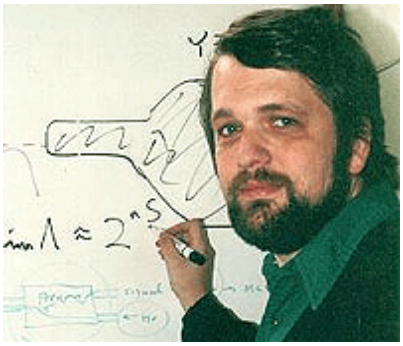
Uno de los obstáculos principales de la computación cuántica es el problema de la decoherencia, que causa la pérdida del carácter unitario (y, más específicamente, la reversibilidad) de los pasos del algoritmo cuántico. Los tiempos de decoherencia para los sistemas candidatos, en particular el tiempo de relajación transversal (en la terminología usada en la tecnología de resonancia magnética nuclear e imaginería por resonancia magnética) está típicamente entre nanosegundos y segundos, a temperaturas bajas.

Las tasas de error son típicamente proporcionales a la razón entre tiempo de operación frente a tiempo de decoherencia, de forma que cualquier operación debe ser completada en un tiempo mucho más corto que el tiempo de decoherencia. Si la tasa de error es lo bastante baja, es posible usar eficazmente la corrección de errores cuántica, con lo cual sí serían posibles tiempos de cálculo más largos que el tiempo de decoherencia y, en principio, arbitrariamente largos. Se cita con frecuencia una tasa de error límite de 10^{-4} por debajo de la cual se supone que sería posible la aplicación eficaz de la corrección de errores cuánticos.

Otro de los problemas principales es la escalabilidad, especialmente teniendo en cuenta el considerable incremento en qubits necesarios para cualquier cálculo que implica la corrección de errores. Para ninguno de los sistemas actualmente propuestos es trivial un diseño capaz de manejar un número suficientemente alto de qubits.

LA UNIDAD DE INFORMACIÓN CUÁNTICA

Seguiremos nuestra discusión introduciendo ahora la unidad elemental de información utilizada en computación cuántica. Al respecto, Benjamín Schumacher –un físico teórico interesado en la teoría cuántica de la información- descubrió, a finales del siglo XX, la forma de interpretar los estados cuánticos como información, y acuñó el término qubit. También descubrió una manera de comprimir la información en un estado y de almacenar la información en el número más pequeño de estados. Este planteamiento no es otra cosa que la analogía cuántica de la teoría de la información clásica de Shannon y, actualmente, se conoce como compresión de Schumacher.

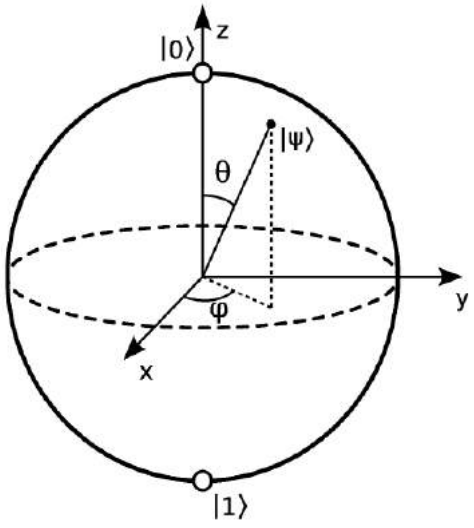


Benjamín Schumacher descubrió una manera de interpretar cuánticamente la información. Se le ocurrió una forma de comprimir la información en un estado, y almacenar la información en un menor número de estados. Esto ahora se conoce como compresión Schumacher. Este fue el análogo cuántico del teorema de codificación sin ruido de Shanonn, y ayudó a iniciar el campo conocido como la teoría de la información cuántica.

El término qubit se atribuye al artículo de Benjamín Schumacher en el que describía una forma de comprimir la información en un estado y de almacenar la información en el número más pequeño de estados, que ahora se conoce como compresión de Schumacher. En el artículo, Schumacher indicó que el término se inventó como broma, por su semejanza fonética con cubit (codo, en inglés), durante una conversación con William Wootters. Posteriormente, por analogía al qubit, se denominó ebit a la unidad para cuantificar entrelazamiento cuántico, y qutrit al análogo del qubit con tres, y no dos, estados cuánticos, representados convencionalmente por: $|0\rangle$, $|1\rangle$ y $|2\rangle$ (kets cero, uno y dos). Para más dimensiones

del espacio de Hilbert, o cuando se está generalizando a “d” dimensiones, se habla de qudit.

Un qubit (del inglés quantum bit o bit cuántico) es un sistema cuántico con dos estados propios y que puede ser manipulado arbitrariamente. Esto es, se trata de un sistema que sólo puede ser descrito correctamente mediante la mecánica cuántica, y solamente tiene dos estados bien distinguibles mediante medidas. También se entiende por qubit la información que contiene ese sistema cuántico de dos estados posibles. En esta acepción, el qubit es la unidad mínima y por lo tanto constitutiva de la teoría de la información cuántica. Es un concepto fundamental para la computación cuántica y para la criptografía cuántica, el análogo cuántico del bit en informática. Su importancia radica en que la cantidad de información contenida en un qubit, y, en particular, la forma en que esta información puede ser manipulada, es fundamental y cualitativamente diferente a la de un bit clásico. Hay operaciones lógicas, por ejemplo, que son posibles en un qubit y no en un bit.



Representación angular de un estado arbitrario de un qubit.

El concepto de qubit es abstracto y no lleva asociado un sistema físico concreto. En la práctica, se han preparado diferentes sistemas físicos que, en ciertas condiciones, pueden describirse como qubits o conjuntos de qubits. Los sistemas pueden ser de tamaño macroscópico -como una muestra de resonancia magnética nuclear o un circuito superconductor-, o microscópico -como un conjunto de iones suspendidos mediante campos eléctricos o los defectos cristalográficos en el diamante-.

Matemáticamente, un qubit puede describirse como un vector de módulo unidad en un espacio vectorial complejo bidimensional. Los dos estados básicos de un qubit son $|0\rangle$ y $|1\rangle$, que corresponden al 0 y 1 del bit clásico (se pronuncian: ket cero y ket uno). Pero además, el qubit puede encontrarse en un estado de superposición cuántica, que es combinación de esos dos estados:

$$|\psi\rangle = a\,|0\rangle + b\,|1\rangle$$

En esto es significativamente distinto al estado de un bit clásico, que puede tomar solamente los valores 0 o 1. Los valores representados por un qubit son de naturaleza continua.

Otra cuestión importante es el paralelismo cuántico, que es la posibilidad de representar simultáneamente los valores 0 y 1. Los algoritmos cuánticos que operan sobre estados de superposición realizan simultáneamente las operaciones sobre todas las combinaciones de las entradas. En este "paralelismo cuántico" reside la potencia del cómputo cuántico.

Una tercera característica importante es que múltiples qubits pueden presentarse en un estado de entrelazamiento cuántico. El entrelazamiento es una característica no local que permite que un sistema de qubits se exprese con una correlación más alta que la posible en sistemas clásicos. Un sistema de dos qubits entrelazados no puede descomponerse en factores independientes para cada uno de los qubits. Este estado puede utilizarse para realizar la teleportación cuántica.

Por último, cualquier sistema cuántico de dos niveles se puede utilizar para representar un qubit. Los sistemas de niveles múltiples se pueden utilizar también, si poseen dos estados que se puedan desemparejar con eficacia del resto (por ejemplo, el estado de tierra y el primer estado excitado de un oscilador no lineal). Hay varias opciones de este tipo de sistemas que se han puesto en práctica con éxito. Además, distintas implementaciones de qubits podrían emplearse juntas para construir un computador cuántico, de la misma forma que se hace en la computación clásica, en donde un bit puede representarse mediante el estado de un transistor en una memoria, por el estado de magnetización de un disco duro o por la transmisión de corriente en un cable.

Elementos y Conceptos de la Computación Cuántica

Se ha argumentado que lo más curioso de la teoría de la información cuántica es el propio concepto de la información cuántica, representado habitualmente por el qubit, y que ésta ofrece una nueva perspectiva a la física, complementaria a la perspectiva geométrica. Es la analogía cuántica de la teoría de la información clásica de Shannon. En la física clásica ya se encontraban relaciones fuertes con la información, como en el caso de la entropía ilustrado por el demonio de Maxwell. En mecánica cuántica esta relación se amplía, y se encuentran resultados como el teorema de no clonación, que impide el copiado de un estado cuántico no conocido, con consecuencias profundas en computación cuántica pero también con una relación clara con el principio de indeterminación. En cualquier caso, vamos a tener que tratar, en mayor o profundidad, los siguientes aspectos:

- Registro cuántico: Varios qubits juntos forman un registro de qubits o registro cuántico. Las computadoras u ordenadores cuánticos ejecutan algoritmos cuánticos, tales como el algoritmo de Shor que descompone en factores un número N con una complejidad computacional menor en tiempo y en espacio, manipulando qubits mediante puertas cuánticas.

- Naturaleza analógica de los qubits: Ya se ha indicado una de las diferencias entre bit y qubit: un bit toma valores discretos mientras que los valores representados por un qubit son de naturaleza continua. Sin embargo, esta característica podría replicarse con magnitudes continuas clásicas (longitudes, voltajes, etc).

- Paralelismo cuántico: El paralelismo cuántico es la posibilidad de representar simultáneamente los valores 0 y 1. Los algoritmos cuánticos que operan sobre estados de superposición realizan simultáneamente las operaciones sobre todas las combinaciones de las entradas. Por ejemplo, los dos qubits:

$$|\psi\rangle=1/2(|0\rangle+|1\rangle)(|0\rangle+|1\rangle)=1/2(|0\rangle|0\rangle+|0\rangle|1\rangle+|1\rangle|0\rangle+|1\rangle|1\rangle)$$

  representan simultáneamente las combinaciones 00, 01, 10 y 11. En este "paralelismo cuántico" reside la potencia del cómputo cuántico.

- Una tercera característica importante que distingue al qubit del bit clásico es que múltiples qubits pueden presentarse en un estado de entrelazamiento cuántico. En el estado no entrelazado siguiente:

$$|\psi\rangle=1/2(|0\rangle|0\rangle+|0\rangle|1\rangle+|1\rangle|0\rangle+|1\rangle|1\rangle)$$

  pueden darse las cuatro posibilidades: que la medida del primer qubit sea 0 ó 1 y que la medida del segundo qubit sea 0 ó 1. Esto es posible porque los dos qubits de la combinación son separables (factorizables), pues la expresión anterior puede escribirse como el producto:

$$|\psi\rangle=1/2(|0\rangle+|1\rangle)(|0\rangle+|1\rangle)$$

El entrelazamiento es una característica no local que permite que un sistema de qubits se exprese con una correlación más alta que la posible en sistemas clásicos. Un sistema de dos qubits entrelazados no puede descomponerse en factores independientes para cada uno de los qubits. Sea, por ejemplo, el entrelazamiento de dos qubits en un estado de Bell:

$$|\beta\rangle=(1/\sqrt{2})(|00\rangle+|11\rangle)$$

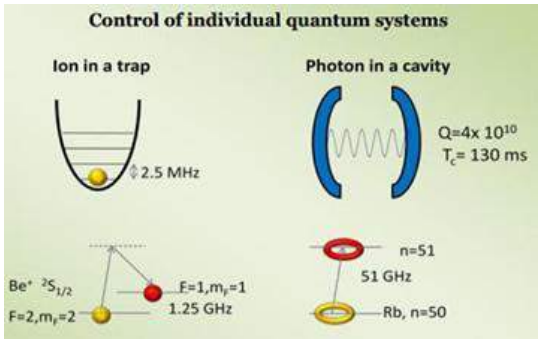
Supongamos que uno de estos dos qubits entrelazados se entrega a Alicia y el otro a Pepe. Alicia hace la medida de su qubit, y supongamos que obtiene el valor 0. Debido al entrelazamiento de los qubits, si Pepe hace ahora su medida, conseguirá el mismo valor que Alicia, es decir, debe obtener 0. Esto es porque no existe el término $|01\rangle$. De la misma forma, si Alicia hace su medida y obtiene el valor 1, y Pepe la hace después, deberá obtener obligatoriamente 1 (puesto que no existe el término $|10\rangle$). De esta forma, el resultado que obtiene Pepe está condicionado por el que obtenga Alicia, aunque estén separados por años luz de distancia. Este estado puede utilizarse para realizar la teleportación cuántica.

- Puertas lógicas cuánticas: Uno de los principales modelos de computación cuántica es el circuito cuántico, en el que se aplican puertas lógicas sobre los qubits. En el modelo de circuito cuántico cualquier algoritmo cuántico se expresa como una serie de puertas lógicas cuánticas que actúan sobre uno o varios qubits. Esta manipulación de los estados cuánticos de dichos qubits incuye la posibilidad de condicionar la aplicación de la puerta lógica del qubit objetivo al estado del qubit control. Un ejemplo típico es la negación controlada. Las puertas lógicas cuánticas tienen ciertas diferencias comparadas con las que se usan en los circuitos digitales convencionales. En particular, todas las puertas lógicas cuánticas son reversibles, es decir, que es posible invertir su acción mediante otra puerta lógica. En la práctica, esto significa que el número de qubits de la entrada ha de coincidir con el de la salida. Cada puerta lógica cuántica se representa por una matriz unitaria.

Representación Física de los Sistemas Computacionales Cuánticos

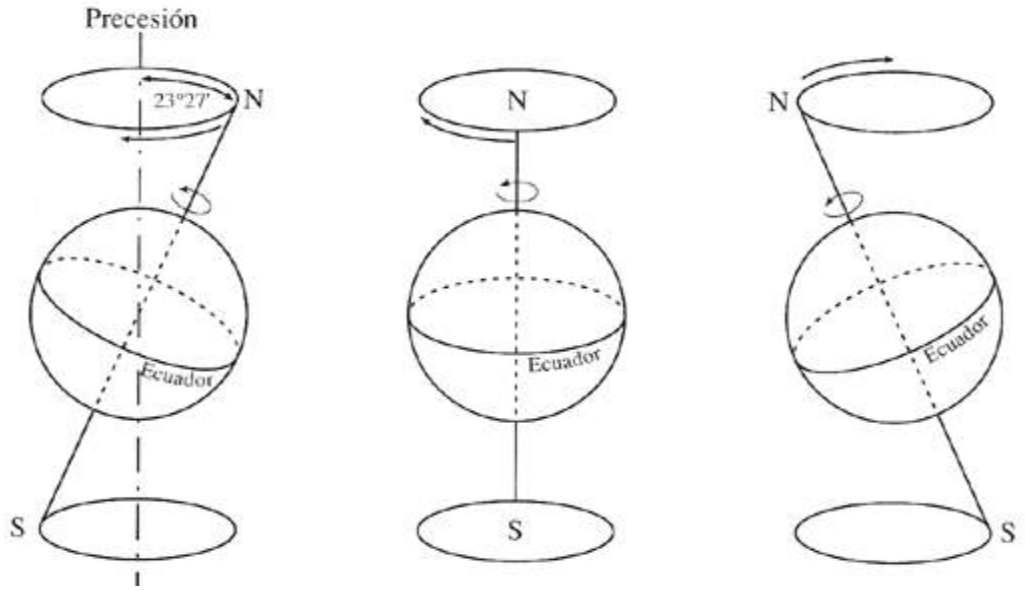
Cualquier estado cuántico de dos niveles se puede utilizar para representar un qubit. Los sistemas de niveles múltiples se pueden utilizar también, si poseen dos estados que se puedan desemparejar con eficacia del resto (por ejemplo, el estado fundamental y el primer estado excitado de un oscilador no lineal). Hay varias opciones de este tipo de sistemas que se han puesto en práctica con diferentes grados de éxito. Por otro lado, distintas implementaciones de qubits podrían emplearse juntas para construir un computador cuántico, de la misma forma que se hace en la computación clásica, en donde un bit puede representarse mediante el estado de un transistor en una memoria, por el estado de magnetización de un disco duro o por la transmisión de corriente en un cable. Algunos métodos ensayados para manipular qubits son los siguientes:

- Trampa de iones o de átomos*:* Si se considera un ion atrapado en una trampa iónica y enfriado mediante láser, es posible considerar como un qubit al estado fundamental y uno de sus estados excitados electrónicos. Se han llevado a cabo experimentos que muestran operaciones elementales de computación en este tipo de sistemas, en los que la interacción de Coulomb actúa como comunicación entre qubits. La manipulación de decenas de iones en ese tipo de trampas conlleva enormes dificultades experimentales; se han hecho propuestas teóricas sobre cómo escalar ese tipo de esquema a un número mayor de qubits, a base de conectar entre sí una serie de trampas, moviendo a los iones entre ellas cuando es necesario para establecer entrelazamiento o puertas lógicas.



Trampa de iones y confinamiento de fotones.

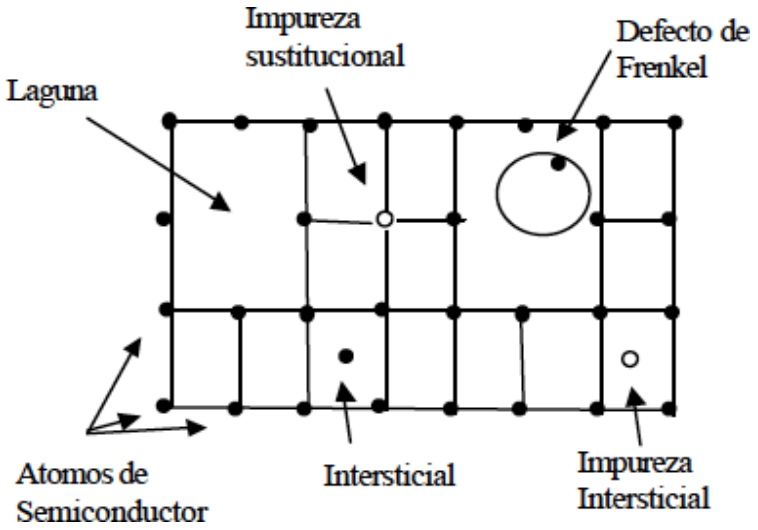
- Espines nucleares*:* El espín de los distintos núcleos atómicos de una molécula sencilla, o, más exactamente, la polarización de la magnetización de esos núcleos en un vasto número de moléculas idénticas puede ser usada como qubits. Varias de las técnicas de resonancia magnética nuclear en disolución que fueron desarrolladas en la segunda mitad del siglo XX pueden ser reinterpretadas en el contexto de la computación cuántica, en concreto algunos de los pulsos de ondas de radio que se usan habitualmente en experimentos sofisticados de elucidación de estructuras químicas se han usado como puertas lógicas cuánticas. En los años 1990 se sucedieron una serie de experimentos de demostración de las bases de la computación cuántica mediante esta implementación. Los primeros resultados fueron espectaculares comparados con otras implementaciones físicas de qubits, pues se beneficiaban de la ciencia y la tecnología de un campo maduro, sin embargo desde entonces el progreso ha sido más lento, principalmente porque el problema de escalar estos experimentos a un número mayor de qubits se encuentra con problemas fundamentales.



Spin nuclear

- Sistemas de estado sólido: Se han llevado a cabo numerosos estudios teóricos e implementaciones experimentales de qubits basados en las uniones de Josephson entre materiales superconductores, que aprovechan las propiedades de los pares de Cooper. En particular, se han preparado y caracterizado superposiciones de estados en anillos superconductores entre corrientes en un sentido y en sentido opuesto. Estas investigaciones se enmarcan en los estudios de las uniones de Josephson como sistemas cuánticos con un número macroscópico de partículas, parte de la exploración de la frontera entre la física clásica y la cuántica.

- Defectos cristalinos en diamante: Entre los muchos posibles defectos cristalográficos de los diamantes se encuentran los pares de nitrógeno-vacante, NV, que consisten en la sustitución de dos átomos de carbono por uno de nitrógeno, quedando una de las posiciones sin ocupar. Por la diferencia de configuración electrónica entre el carbono, que tiene cuatro electrones de valencia y el nitrógeno, que tiene cinco, esto conlleva necesariamente un electrón desapareado.



Distintos tipos de defectos cristalinos.

Sin embargo, el caso que ha sido más explorado es el centro nitrógeno-vacante aniónico, en el que hay un electrón extra ocupando la vacante, con una fuerte interacción de canje que resulta en un estado de espín S=1. Como ese espín presenta un considerable desdoblamiento a campo nulo, el par $m_s = \pm 1$ es lo que puede servir como qubit, y se han llevado a cabo experimentos que muestran el acoplamiento coherente entre dos de estos qubits. También se ha logrado observar dinámicas de espín coherentes entre el espín electrónico y el espín nuclear de algunos de átomos ^{13}C cercanos al centro NV, que pueden considerarse como una memoria, puesto que están relativamente protegidos de la decoherencia.

--...--

POSTULADOS DE LA MECÁNICA CUÁNTICA Y OPERADORES

La teoría clásica de la computación habitualmente no hacía referencia a la física del dispositivo, y se suponía que los fundamentos de tal teoría eran independientes de la realización física de los mismos. Hubieron de pasar 20 años antes de que Deutsch, Feynman y otros pusieran de manifiesto que esta idea era falsa, mostrando la conexión entre las leyes de la física y la información, en concreto con la computación. A partir de aquí se produjo una más de tantas uniones entre ideas distintas que han aparecido en la física: computación y mecánica cuántica. De esta unión surgió la computación cuántica. De forma general podemos decir que la computación es la creación de conjuntos de símbolos (resultados) a partir de ciertos conjuntos de símbolos iniciales (o datos). Si interpretamos los símbolos como objetos físicos, la computación correspondería a la evolución de los estados de los sistemas. Por tanto, dicha evolución es un ejemplo de computación. Si la evolución es cuántica, tenemos la Computación Cuántica.

La posibilidad de que una máquina de Turing cuántica pudiera hacer algo genuinamente cuántico fue planteada por Richard Feynman en 1982, demostrando que ninguna máquina de Turing clásica (probabilista o no) podía simular algunos comportamientos cuánticos sin incurrir en una ralentización exponencial; sin embargo una máquina de Turing cuántica sí podía hacerlo. Este comportamiento surge del hecho de que la dimensión del espacio de Hilbert accesible al sistema aumenta de forma exponencial con el número de amplitudes a manejar y guardar. Feynman describió un "simulador cuántico universal" que simulaba el comportamiento de cualquier sistema físico finito.

Dado que la computación cuántica se basa en las propiedades cuánticas de los qubits, y que la manipulación de qubits debe realizarse de acuerdo con las leyes y restricciones de la mecánica cuántica, antes de seguir con el desarrollo de este nuevo paradigma computacional, conviene dar una ligera capa de barniz sobre los principios básicos y el lenguaje de la mecánica cuántica.

Postulados de la Mecánica Cuántica

La computación cuántica es un tipo de computación que utiliza las herramientas de la mecánica cuántica para su desarrollo. Por esta razón, y aunque no es el propósito de este texto dar conocer en profundidad los entresijos de la mecánica cuántica, sí que va a ser necesario dominar con cierta soltura algunos de sus principios más básicos y elementales.

En este contexto, a comienzos del siglo XX los físicos no podían describir correctamente el comportamiento de partículas muy pequeñas como electrones, núcleos de átomos y moléculas. El comportamiento de dichas partículas se describe correctamente con un conjunto de leyes físicas que denominamos Mecánica Cuántica.

A principios del siglo pasado, un reducido número de físicos, entre los que podemos citar Bohr, Einstein, Born, Dirac, Schrödinger, Heisember, De Broglie, Jordan, Pauli, contribuyeron a formalizar matemáticamente la Teoría que quedó prácticamente completa a finales de la década de 1920.

Foto de familia de la decisiva conferencia Solvay

El estudio de la Mecánica Cuántica se puede realizar siguiendo dos caminos diferentes. La primera vía consiste en analizar aquellos problemas físicos que la Mecánica Clásica es incapaz de resolver y que, sin embargo, fueron interpretados correctamente por la Mecánica Cuántica. Podemos citar:

- La Ley de radiación espectral del cuerpo negro.
- El efecto fotoeléctrico.
- Las capacidades caloríficas de los sólidos.
- El espectro atómico del átomo de hidrógeno.
- El efecto Compton.

La segunda vía que podemos seguir es la axiomática. Partimos de unos postulados fundamentales a partir de los cuales se deducen resultados sobre el comportamiento de los sistemas físicos microscópicos. Estos resultados se contrastan con el experimento pudiéndose observar el mayor o menor acuerdo entre la teoría y los datos experimentales, lo que proporciona una medida directa de la bondad de la teoría.

En este texto abordaremos el estudio de la Mecánica Cuántica desde el punto de vista axiomático. Las formulaciones más conocidas son el formalismo de Schrödinger que se basa en la descripción ondulatoria de la materia. Por otra parte, el formalismo de Heisenberg y Dirac, que es el que presentaremos aquí, emplea algebra de vectores, operadores y matrices. Schrödinger demostró que ambos formalismos son equivalentes y pueden utilizarse indistintamente.

La formulación axiomática de Heisenberg y Dirac se basa en los postulados que enunciamos a continuación:

- **Postulado I.** El estado de un sistema físico está descrito por una función $\Psi(q,t)$ de las coordenadas (q) y del tiempo (t). Esta función, llamada función de estado o función de onda, contiene toda la información que es posible determinar acerca del sistema. Además, postulamos que

$\Psi(q,t)$ toma valores simples, es finita, continua, con derivadas continuas y de cuadrado integrable.

- **Postulado II.** La evolución en el tiempo del estado de un sistema está dada por la ecuación de Schrödinger dependiente del tiempo:

$$i\hbar\frac{\partial\Psi(q,t)}{\partial t} = \mathrm{H}\Psi(q,t)$$

Donde $\hbar=h/2\pi$, siendo h una constante universal conocida como constante de Planck, y donde **H** es el operador de Hamilton (o Hamiltoniano) del sistema. Para una única partícula moviéndose a lo largo del eje x, **H** viene dado por:

$$\mathrm{H} = -\frac{\hbar^2}{2m}\frac{\partial\Psi(q,t)}{\partial x} + V(x,t)\Psi(x,t)$$

- **Postulado III.-** A cada observable físico en mecánica cuántica le corresponde un operador lineal y hermítico. Para encontrar dicho operador, escribimos la expresión clásica del observable en términos de las coordenadas cartesianas y de los momentos lineales correspondientes. A continuación, reemplazamos cada coordenada x por el operador $\boldsymbol{x}$ (multiplica por x) y cada momento lineal p_x por el operador $-i\hbar\partial/\partial x$.

- **Postulado IV.-** Independientemente de cuál sea la función de estado de un sistema, los únicos valores que pueden resultar de una medida del observable físico A son los valores propios "a", de la ecuación: $\boldsymbol{A}f_i=af_i$

- **Postulado V.-** Si $\boldsymbol{A}$ es un operador hermítico lineal que representa un observable físico, entonces las funciones propias ψi de la ecuación de valores propios $A\psi_i=a_i\psi_i$, forman un conjunto completo. Esto quiere decir que cualquier función de estado Ψ que satisfaga las mismas

condiciones límite que cada ψ_i puede expresarse como combinación lineal de los estados propios de ***A***.

$$\Psi = \sum c_i \psi_i$$

- **Postulado VI.-** Si $\psi_i(q,t)$ es la función de estado normalizada de un sistema al tiempo t, entonces el valor medio de un observable físico A en el instante t es:

$$\langle A \rangle = \int \psi^* A \psi dq$$

- **Postulado VII.-** Si ***A*** es un operador lineal y hermítico que respresenta a un observable físico, las funciones propias, f_i, del operador ***A*** forman un conjunto completo.

Operadores

Un operador es una regla o procedimiento que permite, dada una función, calcular otra función correspondiente. Lo que sigue a continuación es una lista de cosas, operaciones, y propiedades, protagonizadas por nuestros amigos los operadores cuánticos. Así, podemos definir las siguientes operaciones básicas de los operadores cuánticos, que denotaremos mediante letras mayúsculas.

- Suma: (A + E) f(x) = A f(x) + E f(x)
- Producto: (A × E) f(x) = A { E f(x)}

Hay que tener en cuenta que, generalmente, (A × E) f(x) ≠ (E × A) f(x). También podemos definir un álgebra de operadores cuánticos con los siguientes elementos:

- Dados dos operadores A y E : A = E $\leftrightarrow$ A f = E f , $\forall$ f
- El operador unidad está definido

- El operador nulo está definido
- Se cumple la propiedad asociativa: $A\ (E\ I) = (A\ E)\ I$
- No siempre se cumple la propiedad conmutativa: $A\ E \neq E\ A$
- Se define el conmutador como: $[A\ ,\ E] = A\ E - E\ A$
- Se define el cuadrado de un operador como: $A^2 = A\ A$

Además, los operadores cuánticos tienen ciertas propiedades interesantes, entre las que destacamos las siguientes:

1. $A\ [\ f(x) + g(x)\ ] = A\ f(x) + A\ g(x)$ donde f y g son funciones
2. Si f es una función, y c es una constante: $A\ [c\ f(x)] = c\ A\ f(x)$
3. $(A + E\ )\ I = A\ I + E\ I$
4. $A\ (E + I) = A\ E + A\ I$

"A" es un operador lineal si y sólo si cumple las propiedades (1) y (2) que acabamos de destacar. En la aproximación cuántica todos los operadores son lineales.

Otro aspecto importante de los operadores cuánticos tiene que ver con los conceptos de funciones propias y de valores propios que comentamos a continuación:

Sean A un operador, f(x) una función, y k una constante. En este caso, si

$$A\ f(x) = k \times f(x)$$

entonces f(x) es una función propia del operador A, y k es un valor propio. Además, las funciones propias de cualquier operador lineal contienen una constante multiplicativa arbitraria c. Efectivamente, sabemos que

$$A\ (c \times f) = c \times A\ f = c \times k \times f = k\ (c \times f)$$

Así, si f(x) es una función propia de A con valor propio k, también c × f(x) es una función propia. Si para cada valor diferente de k tenemos una función propia distinta, y funciones propias con el mismo valor de k pero valores distintos de c no son independientes entre sí.

También relacionada con los operadores cuánticos está la cuestión de los valores promedio, que utilizaremos un poco más adelante, es este mismo texto. Consideremos una magnitud física E. Cuando la función de estado Ψ no es una función propia del operador asociado "E", una medida de E nos da uno de entre un número de valores posibles. Consideremos ahora el valor promedio de la propiedad E para un sistema cuyo estado sea Ψ. La determinación experimental del valor promedio de E necesita considerar un gran número de sistemas, todos ellos en el mismo estado Ψ y efectuar una medida de E en cada sistema. Así, el valor promedio de E se define como la media aritmética de los valores observados. Si e1, e2,… son los valores observados de E, el valor promedio de E -que denotamos ⟨E⟩- para un número N de sistemas muy grande, es el siguiente:

$$\langle E \rangle = \frac{\sum_{i=1}^{N} e_i}{N}$$

El mismo resultado se obtiene sumando todos los posibles valores de E (los distintos e), multiplicando cada valor posible por el número de veces que dicho valor ha sido observado:

$$\langle E \rangle = \frac{\sum_{e} n_e \times e}{N} = \sum_{e} \left(\frac{n_e}{N} \right) \times e = \sum_{e} P_e \times e$$

En esta última expresión, P_e es la probabilidad de observar el valor "e", ya que, como hemos dicho, N es muy grande. Podemos utilizar estos resultados para estudiar el comportamiento de un sistema unidimensional de una partícula en el estado Ψ(x,t). Concretamente consideraremos el valor promedio de su coordenada x. Sea una partícula |bit| en el estado Ψ(x, t) de tal forma que |x| toma valores en

un continuo. Entonces $P_{bit}|x, x + dx| = |\Psi|^2 dx$ es la probabilidad de observar a la partícula entre (x) y (x+dx). Por lo tanto:

$$\langle x \rangle = \int_{-\infty}^{+\infty} x \,|\, \Psi(x,t) |^2 \; dx$$

Por otra parte, si E(x) es una propiedad que depende de x, y si "E" es el operador asociado a E(x), entonces:

$$\langle E \rangle = \int \Psi * E \Psi dx$$

La densidad de probabilidad se define como: $\Psi * \Psi$. Además, si F y G son dos propiedades cualesquiera, entonces:

$$\langle F + G \rangle = \langle F \rangle + \langle G \rangle$$

$$\langle F \times G \rangle \neq \langle F \rangle \times \langle G \rangle$$

Un inconveniente de todo este jaleo es que la notación con integrales puede llegar a ser engorrosa. La solución viene de la mano de Paul Dirac, quien propuso una notación alternativa, que esbozamos a continuación:

Si τ indica todo el espacio de integración, si φ m y φ n son funciones, y si "A" es un operador, entonces:

$$\int \varphi_m * A \varphi_n d\tau \equiv \langle \varphi_m | A | \varphi_n \rangle \equiv (\varphi_m | A | \varphi_n) \equiv \langle m \,|\, A \,|\, n \rangle \equiv A_{mn}$$

$$\int \varphi_m * \varphi_n d\tau \equiv \langle \varphi_m \,|\, \varphi_n \rangle \equiv \langle m \,|\, n \rangle$$

Por otra parte, si un estado dado puede ser descrito como un vector columna utilizaremos la notación 'ket', y si puede ser descrito como un vector fila utilizaremos la notación 'bra' tal y como se ilustra a continuación:

$$\text{ket}\,(\psi) = |\psi\rangle = \begin{pmatrix} a \\ b \end{pmatrix} \qquad \text{y} \qquad \text{bra}\,(\psi) = \langle \psi | = (a \; b)$$

Además se cumple que:

$$\langle m|n\rangle^* = \langle n|m\rangle$$

$$\langle m|m\rangle^* = \langle m|m\rangle$$

Ya mencionamos que no es el propósito de este texto servir de tratado de mecánica cuántica. Por ello, creemos que las nociones aquí presentadas deben ser suficientes para entender lo que pretendemos explicar de computación cuántica. Queda, no obstante, un asunto pendiente. Y es que en mecánica cuántica las operaciones son reversibles. Como tenemos que operar según las limitaciones que la mecánica cuántica impone, esta circunstancia nos lleva a tratar formalmente el problema de la reversibilidad. Será inmediatamente.

--..--

COMPUTACIÓN REVERSIBLE Y COMPUTACIÓN CUÁNTICA

Se puede clasificar los procesos de cómputo en aquéllos para los que lo fundamental es una operación irreversible o disipativa, y en los que para los cuales esta circunstancia no es fundamental. Una compuerta lógica cuya información de salida sea menor que la de entrada es irreversible, pues tuvo que desechar información, lo cual se traduce en última instancia en una pérdida de energía de algún tipo. Por el contrario, una compuerta lógica cuya información de salida sea igual que la información de entrada será reversible o no disipativa, y la información permanece "constante", lo cual lleva consigo una energía también constante. Una compuerta de este tipo, o mejor, un sistema de compuertas de este estilo, conectadas de alguna forma para que sea capaz de realizar una operación lógica, es susceptible de que en ella se invierta el proceso de cómputo y al final se recuperen las condiciones iniciales sin pérdida alguna de energía.

La idea de computación clásica reversible la introdujo matemáticamente Yves Lecerf en 1963 y la desarrolló Bennett en 1973 demostrando que, desde un punto de vista teórico, es posible la existencia de una máquina de Turing reversible.

C. H. Bennett (1943) es Físico de IBM. Ha jugado un papel principal en la comprensión de las interconexiones entre la física y la información, en particular en el reino de cómputo cuántico, pero también en autómatas celulares y en la informática reversible. Descubrió, con Gilles Brassard, el concepto de criptografía cuántica y es uno de los padres fundadores de teoría moderna cuántica de la información.

La existencia de tales máquinas de Turing reversibles nos indica que no hay una cantidad mínima de energía que haya que poner en juego para efectuar un cómputo concreto. En este contexto, Bennett demostró que las compuertas irreversibles no son esenciales en los procesos de cómputo. Aún más, la

reversibilidad es fundamental en la computación cuántica. Si queremos construir máquinas cuánticas debemos respetar las leyes de la física cuántica, pero resulta que tales leyes son reversibles en el tiempo, lo que nos obliga a realizar operaciones reversibles.

Introduciendo la Reversibilidad

Supongamos que queremos sumar dos números binarios… digamos el número binario |001101| con el |010110|. Evidentemente el resultado es |100011|. Podemos visualizar esto considerando cintas que transportan cajas, con bolas o sin bolas, de acuerdo con las siguientes restricciones:

1. El número de cajas de la cinta de ambos sumandos debe ser el mismo.
2. El número de cajas de la cinta de ambos sumandos debe ser, por lo menos, igual al número de bits de la mayor palabra1 que queremos sumar.
3. Colocaremos el resultado de la suma en las cajas de una tercera cinta.
4. En esta tercera cinta debe haber, al menos, una caja más que las que hay en cualquiera de las cintas que transportan a nuestros sumandos.
5. En cualquier cinta, cada caja puede contener una bola -|1⟩- o ninguna -|0⟩-
6. La suma se efectúa vaciando, en cada posición y empezando por la derecha, el contenido de los sumandos, en la caja correspondiente de la cinta del resultado.
7. Si por cualquier razón la mercancía no cabe en la caja correspondiente de la cinta del resultado, se transfiere la mercancía a la caja que está inmediatamente a su izquierda hasta que quepa.
8. El proceso se repite hasta que ya no haya nada más que sumar.

Con esto podemos establecer las reglas básicas de la suma binaria de dos bis, A y B, que detallamos a continuación.

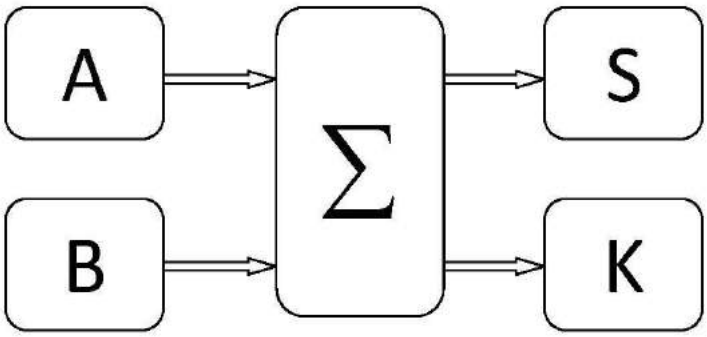
$\|A\rangle$	$\|B\rangle$	$\|S\rangle$	$\|K\rangle$
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Reglas básicas de la suma binaria de dos bis A y B.

Si recordamos lo que ya sabemos sobre operaciones lógicas, resulta que:

$$(A \wedge B) = 1 \leftrightarrow |A\rangle = |B\rangle = 1$$

y podemos comprobar que el bit de acarreo del semisumador y la conjunción binaria son una misma cosa. De este modo, el bit de acarreo «K» del semisumador se puede obtener directamente introduciendo las señales A y B como entradas de una puerta lógica AND. Completamos el razonamiento: ¿Cómo podemos obtener el bit suma del sumador «S» utilizando otra operación lógica?



Representación esquemática de un semisumador con entradas A y B y salidas Suma (S) y acarreo (K)

Volviendo a las operaciones lógicas, recordamos que:

$$|A\rangle = 1 \wedge |B\rangle = 0 \rightarrow (A \oplus B) = 1$$

$$|A\rangle = 0 \wedge |B\rangle = 1 \rightarrow (A \oplus B) = 1$$

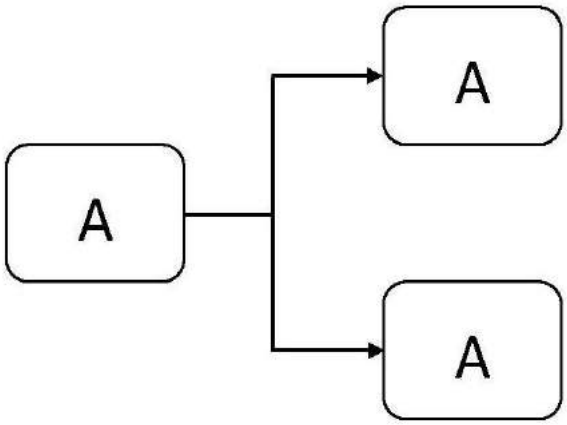
$$|A\rangle = 1 \wedge |B\rangle = 1 \rightarrow (A \oplus B) = 0$$

$$|A\rangle = 0 \wedge |B\rangle = 0 \rightarrow (A \oplus B) = 0$$

Por lo tanto, la disyunción exclusiva $(A \oplus B)$ nos suministra el bit suma del semisumador.

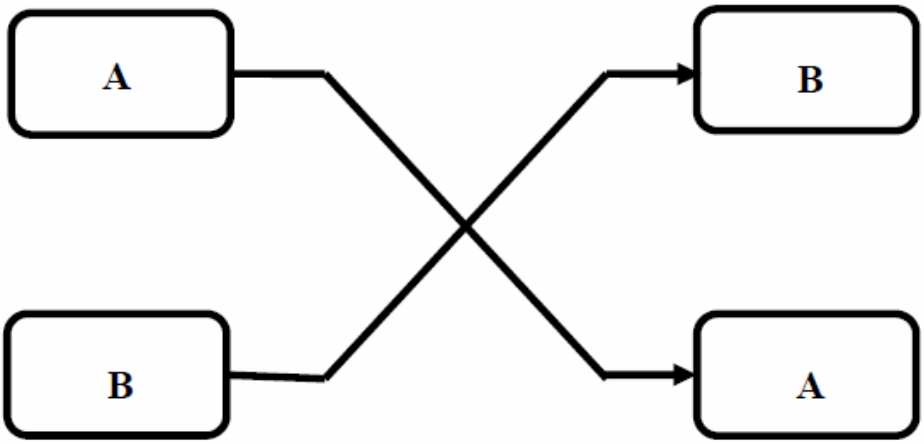
Operaciones Reversibles

La conjunción, la disyunción, y la disyunción exclusiva binarias pueden englobarse en lo que habitualmente se denominan funciones de conmutación. Las funciones de conmutación aceptan como entrada algunas variables binarias y calculan alguna función binaria. Además, sabemos que el conjunto de operadores binarios {AND, OR, NOT} es un conjunto completo, por medio de cuyos elementos puede, en principio, construirse cualquier función lógica. También sabemos que existen operadores que por sí solos forman un conjunto completo, por ejemplo {NAND} y {NOR}. Estamos ya casi a punto de poder discutir sobre reversibilidad, pero antes vamos a definir dos operaciones lógicas nuevas, la bifurcación, FO o BIF, y el intercambio, EX, de las que haremos uso cuando hablemos, por fin, de computación reversible. La operación de bifurcación «FO» divide una entrada en dos o más.



Una operación de bifurcación o "fanout", FO.

Por otra parte la operación de intercambio «EX» simplemente intercambia el par de conexiones de la entrada.



Una operación de intercambio o "exchange", EX.

Estas dos operaciones obvias van a ser necesarias para discutir sobre la reversibilidad, y además supondremos que tenemos a nuestra disposición un número suficiente de bits $|0\rangle$ y de bits $|1\rangle$ durante todo el tiempo que deseemos.

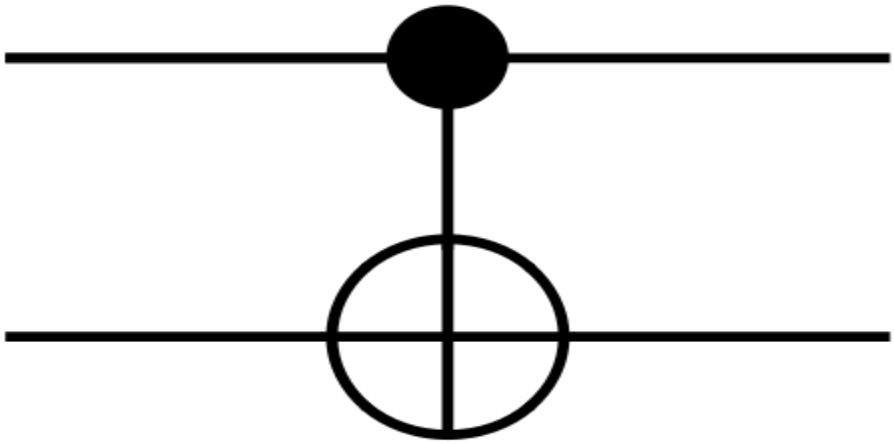
Fijémonos ahora en las operaciones AND, NAND, OR, XOR. Estas operaciones son irreversibles, en el sentido de que a partir de la salida no se puede reconstruir la entrada. Con operaciones irreversibles perdemos información de forma irreversible.

Definiremos operación reversible como aquella operación lógica que tiene la suficiente información en la salida como para poder reconstruir la entrada. Como aspecto curioso mencionaremos aquí, y desarrollaremos luego, que la reversibilidad es imprescindible para estudiar cuestiones relacionadas con la termodinámica de la computación, ya que nos permite realizar cálculos de energía libre, y conocer la eficiencia física de la computación. Al respecto, fueron Bennet y Fredkin los primeros que, independientemente, estudiaron la posibilidad de construir computadoras reversibles, para lo cual se requieren puertas lógicas reversibles. La primera operación lógica reversible con que nos encontramos es la negación binaria, que se realiza mediante el operador lógico NOT, que podremos denotar a partir de ahora también con la letra N. La negación binaria es claramente reversible. Para comprobarlo basta con recordar que:

$$|1\rangle = \neg\, |0\rangle$$

$$|0\rangle = \neg\, |1\rangle$$

Parecida a la puerta N, vamos a construir ahora la puerta «CN» o disyunción binaria controlada. Esta nueva puerta «CN» es el dispositivo de dos entradas y dos salidas siguiente:



Puerta Controlled-Not, CN. La línea superior es la línea de control. La línea inferior es un NOT controlado por el estado de la línea superior. Esta puerta puede interpretarse como una disyunción binaria de dos entradas y dos salidas.

El funcionamiento de CN debe respetar las siguientes restricciones:

1. $|A_{out}\rangle = |A_{in}\rangle$
2. $|B_{out}\rangle = |B_{in}\rangle \leftrightarrow |A\rangle = 0$
3. $|B_{out}\rangle = \neg\, |B\rangle \leftrightarrow |A\rangle = 1$

Si el estado de la línea (A) es $|1\rangle$ el valor de la entrada de la línea (B) se invierte, pero si la entrada a la línea (A) es $|0\rangle$ entonces la línea (B) pasa sin modificarse. La entrada en la línea (A) activa una puerta N en la línea inferior (B), y la salida (A) es siempre la misma que la entrada (A). Por lo tanto, la tabla de verdad de la puerta CN será la que se muestra a continuación:

| A ⟩	| B ⟩	| A' ⟩	| B' ⟩
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

Tabla de verdad de la puerta CN.

Claramente se puede interpretar $|B_{out}\rangle$ como la salida de una puerta XOR con entradas $|A_{in}\rangle$ y $|B_{in}\rangle$: $|B_{out}\rangle$ = XOR (A_{in},B_{in})... sin embargo el dispositivo no es el mismo ya que la puerta CN genera dos salidas en lugar de una. Esta puerta es perfectamente reversible ya que, una vez conocida la salida, siempre podemos reproducir la entrada. Podemos comprobar la reversibilidad de CN sin más que repetir la operación:

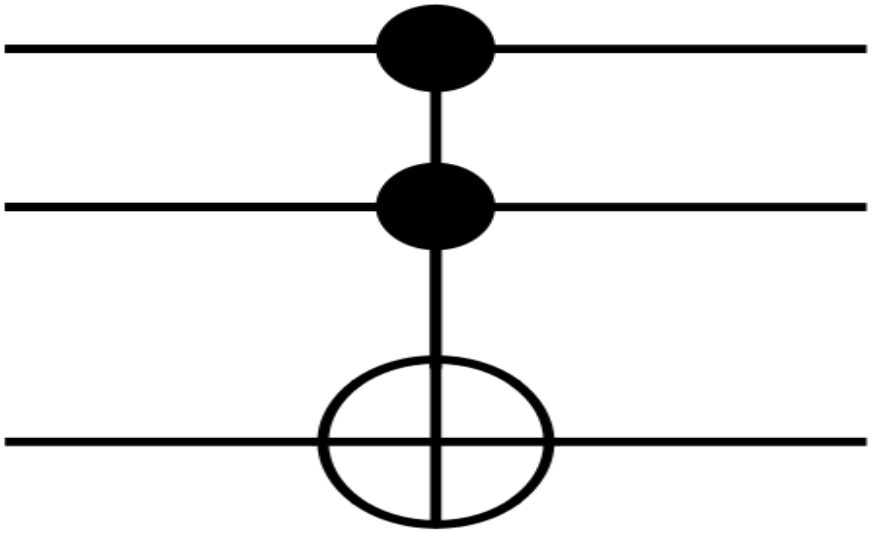
| A ⟩	| B ⟩	| A' ⟩	| B' ⟩	| A'' ⟩	| B'' ⟩
0	0	0	0	0	0
0	1	0	1	0	1
1	0	1	1	1	0
1	1	1	0	1	1

Reversibilidad de CN.

Sin embargo las puertas N y CN no bastan para «hacerlo todo». Necesitamos complicar algo más nuestras puertas reversibles para conseguir una que constituya, por sí sola, un conjunto completo de operadores.

La Disyunción Binaria Doblemente Controlada

La puerta CCN es el operador reversible de la disyunción binaria doblemente controlada, y constituye además un conjunto completo de operadores binarios reversibles.



Disyunción binaria reversible doblemente controlada o puerta Controlled-Controlled-NOT, CCN.

El funcionamiento de la puerta CCN es el siguiente:

1. Hay 2 líneas de control, A y B / $|A'\rangle = |A\rangle$, $|B'\rangle = |B\rangle$
2. La línea C sólo es activada cuando ($|A\rangle = 1$) $\wedge$ ($|B\rangle = 1$)
3. En este último caso $|C'\rangle = \neg |C\rangle$
4. Si mantenemos $|A\rangle = |B\rangle = 1$: CCN se comporta como N en su línea C
5. Si sólo mantenemos $|A\rangle = 1$: CCN se comporta como CN

| A ⟩	| B ⟩	| C ⟩	| A' ⟩	| B' ⟩	| C' ⟩
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
0	1	1	0	1	1
1	0	1	1	0	1
1	1	0	1	1	1
1	1	1	1	1	0

Tabla de verdad de la puerta CCN.

Como antes, comprobaremos la reversibilidad de CCN repitiendo la operación.

| A ⟩	| B ⟩	| C ⟩	| A' ⟩	| B' ⟩	| C' ⟩	| A'' ⟩	| B'' ⟩	| C'' ⟩
0	0	0	0	0	0	0	0	0
0	0	1	0	0	1	0	0	1
0	1	0	0	1	0	0	1	0
1	0	0	1	0	0	1	0	0
0	1	1	0	1	1	0	1	1
1	0	1	1	0	1	1	0	1
1	1	0	1	1	1	1	1	0
1	1	1	1	1	0	1	1	1

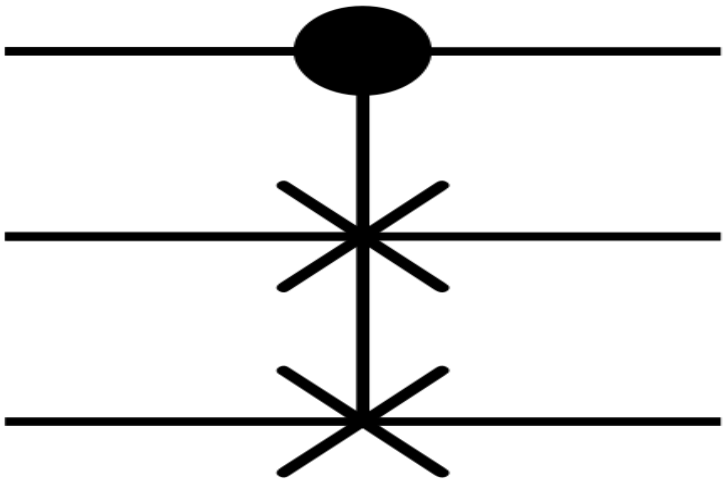
Reversibilidad de CCN.

Ya hemos dicho varias veces que la puerta CCN es por sí misma un conjunto completo de operadores. En esta sección configuraremos esta puerta para que funcione como otras puertas lógicas de las ya estudiadas. Sólo configuraremos algunas. Las que falten las dejaremos como ejercicio para el lector estudioso.

Por ejemplo, el operador lógico AND se puede construir a partir de CCN fijando $|C\rangle = 0$ y alimentando la puerta con A y B. Por otra parte, l operador lógico NAND se puede construir a partir de la puerta CCN fijando $|C\rangle = 1$ y alimentando la puerta con A y B. Finalmente, el operador lógico XOR se puede construir a partir de CCN fijando $|A\rangle = 1$ ó $|B\rangle = 1$ y alimentando la puerta normalmente.

La Puerta de Fredkin

Otra puerta de gran interés en computación reversible es la puerta de Fredkin. Esta puerta reversible introduce un elemento que realiza un intercambio controlado, y en ella el número de $|1\rangle$ y de $|0\rangle$ no cambia nunca.



Puerta de Fredkin o doble negación controlada por la línea superior. Funciona como un intercambio controlado.

El funcionamiento de la puerta de Fredkin es el siguiente:

1. $|A'\rangle = |A\rangle$
2. $|A\rangle = 0 \rightarrow \{ |B'\rangle = |B\rangle \} \wedge \{ |C'\rangle = |C\rangle \}$
3. $|A\rangle = 1 \rightarrow \{ |B'\rangle = | C\rangle \} \wedge \{ |C'\rangle = |B\rangle \}$

Construir la tabla booleana de la puerta de Fredkin y comprobar que con este dispositivo el número de $|1\rangle$ y de $|0\rangle$ no cambia nunca es un ejercicio divertido que recomendamos.

$\|A\rangle$	$\|B\rangle$	$\|C\rangle$	$\|A'\rangle$	$\|B'\rangle$	$\|C'\rangle$
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	1	0
1	1	0	1	0	1
1	1	1	1	1	1

Tabla de verdad de la puerta de Fredkin.

Computadores Reversibles

Con lo visto hasta ahora ya podemos construir un computador reversible. De hecho ya hemos trabajado con algo que era capaz de sumar números de un bit. Diseñaremos ahora un sumador completo reversible, con todo lo que ello significa: conoceremos el resultado de la operación, y los sumandos correspondientes.

Si queremos un sumador reversible necesitamos más información en la salida. En concreto, para resolver el problema planteado necesitamos dos líneas extra que salgan de la puerta y una línea extra en la entrada que configuramos con un valor fijo, por ejemplo $|0\rangle$. El procedimiento, ahora, podría ser el siguiente:

1. Utilizar N, CN, CCN (o sólo CCN, que como ya sabemos es un conjunto completo de operadores)

2. Construir AND, OR, XOR, con los que se puede construir un sumador
3. Utilizar la redundancia de las salidas extra
4. Organizar el sistema de forma que las dos líneas extra, aparte de las salidas de suma «S» y acarreo «K», sean precisamente «A» y «B»

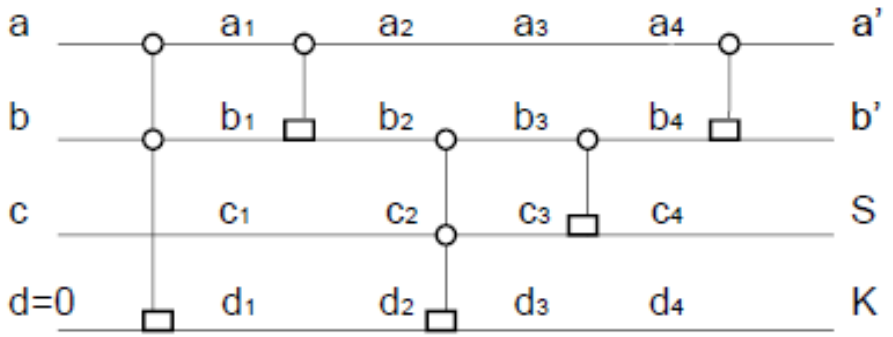
Definiremos computador reversible como aquél que da como salida el resultado real de una computación y además la entrada original. Un aspecto sumamente interesante de la computación reversible, sobre la que discutiremos en profundidad más adelante, es que se puede demostrar que la computación reversible se puede realizar con un coste nulo de energía. El único coste energético en que se incurre aparece cuando reiniciamos la máquina para volver a empezar otra computación. Además, esta reiniciación no depende de la complejidad del cálculo. Sólo depende del número de bits de la respuesta, de forma que se pueden tener N componentes funcionando en una máquina, pero si la respuesta que se obtiene es de sólo un bit, la energía que se necesita para que todo funcione es

$$k \times T \times \ln 2.$$

Por lo tanto se puede afirmar que la computación reversible no necesita el establecimiento de requisitos mínimos de energía.

Por ejemplo el sumador reversible de dos bits de la figura responde a la estructura lógica reversible siguiente:

$$\left\langle \sum \oplus \right\rangle = [a,b,c,d=0]\left\langle CCN(d)_{a,b} \right\rangle | \left\langle CN(b)_a \right\rangle | \left\langle CCN(d)_{b,c} \right\rangle | \left\langle CN(c)_b \right\rangle | \left\langle CN(b)_a \right\rangle$$



Arquitectura básica de un sumador completo de números de dos bits.

La expresión anterior hay que interpretarla del siguiente modo:

1. Configurar un sistema con 4 líneas (a, b, c, d), en el que -d=0-
2. Realizar una operación NOT sobre la línea -d-, doblemente controlada por las líneas -a- y -b-
3. A continuación realizar un NOT sobre -b- controlado por -a-
4. Repetir un NOT sobre -d- pero controlado ahora por -b- y -c-
5. Realizar un NOT sobre -c- controlado por -b-
6. Terminar realizando un NOT sobre -b- controlado por -a-

Si nos fijamos, cualquier computación reversible tiene que almacenar mucha información, desde luego mucha más que una computación convencional. Algo de esto habíamos comentado ya. Una parte de esta información necesaria es el resultado de la propia computación. El resto es, simplemente, la información necesaria para conseguir que nuestra computación sea reversible.

Para que una computación sea útil y eficaz, y al mismo tiempo reversible, nuestro computador debe verificar una serie de restricciones importantes, que tienen que ver con la dirección del proceso de computación. Así, cuando utilizamos un computador convencional, y realizamos un paso hacia delante, no puede haber ninguna ambigüedad.

De forma análoga, una máquina que funcione de acuerdo con los principios de la reversibilidad tampoco puede presentar ninguna ambigüedad en los pasos que realice hacia atrás. Esta característica tiene como consecuencia que la computación reversible es radical y esencialmente diferente de la computación irreversible convencional.

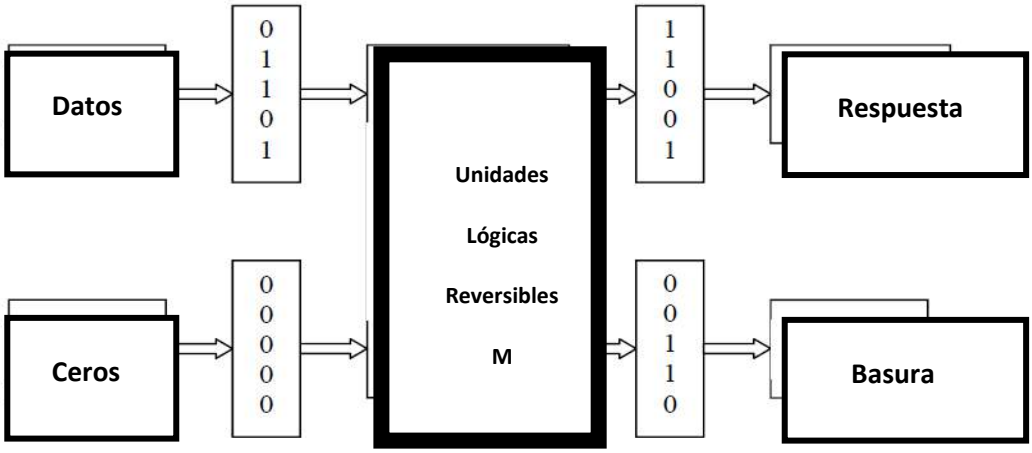
Generalizando el Computador Reversible

Supongamos que tenemos un sistema de unidades lógicas reversibles que están conectadas entre sí para que hagan algo interesante, por supuesto de forma reversible. Introducimos ahora un dato de entrada con el que queremos trabajar para conseguir ese algo que nos fascina y nos parece tan interesante.

Para poder controlar nuestra computación necesitaremos un determinado número de bits $|0\rangle$. Dejamos ahora que nuestro sistema lógico reversible empiece a trabajar con la información de entrada suministrada. Cuando nuestra máquina termine de computar nos dará su resultado, que será:

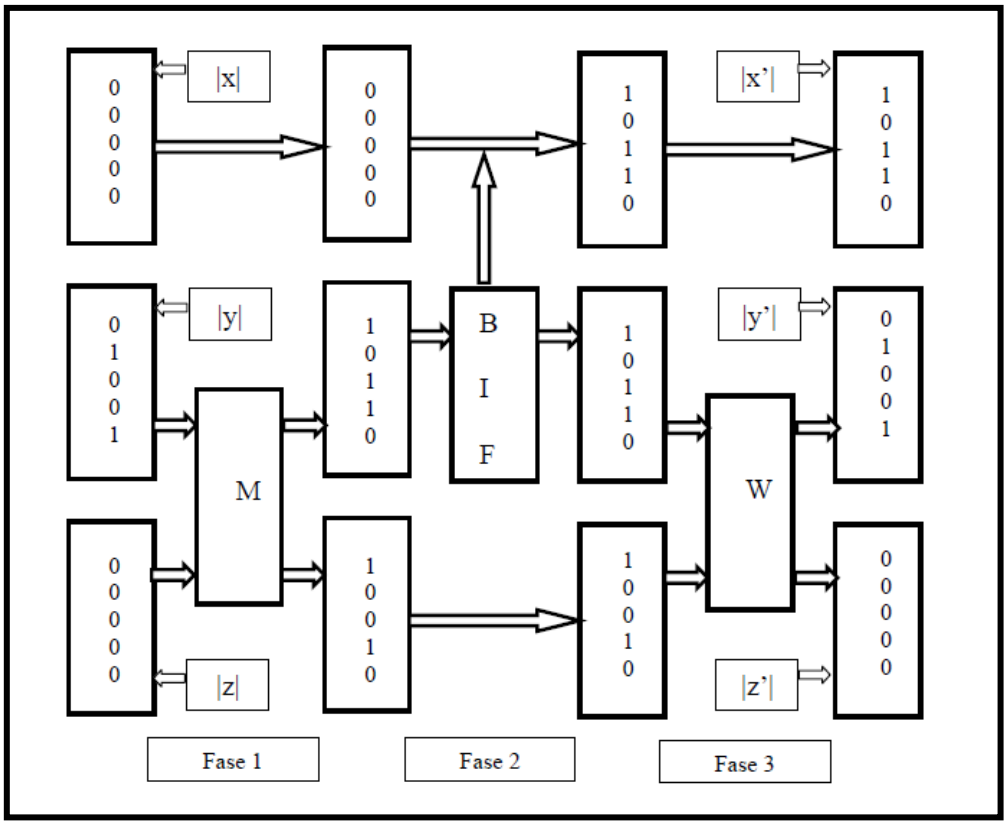
1. La respuesta deseada
2. Un montón de bits sobrantes que encierran la historia del proceso

Dicho así no parece que hayamos hecho gran cosa, pero si nos organizamos bien comprobaremos que se plantean situaciones de suma importancia, relativas a lo que a mí me gusta llamar computación recreativa. Analicemos la arquitectura de la figura.



Arquitectura básica reversible.

En nuestra máquina empezamos con una situación en la que lo sabemos todo: la información de entrada «datos» y los «ceros» de control. Pero terminamos con algo completamente amorfo, reversible pero amorfo. Desde luego es reversible, basta con aplicar la secuencia lógica del sistema $\langle M \rangle$ al revés [es decir $\langle M^{-1} \rangle$] para reproducir la entrada, pero el resultado viene acompañado de mucho desorden. Lo curioso del caso es que el desorden generado puede ser utilizado para mover físicamente una máquina, al menos en teoría. Sin embargo, y como somos muy limpitos, la pregunta que debemos responder ahora es… ¿No podríamos hacer lo mismo sin generar tanta basura? Pues bien, la respuesta es que sí se puede. Para entender cómo, sólo tenemos que recordar nuestra definición de computador reversible: Un computador reversible es aquél que da como salida el resultado real de una computación y además la entrada original. Evidentemente, la solución pasa por incluir en nuestra arquitectura la secuencia de operaciones lógicas reversibles $\langle M^{-1} \rangle$. Vamos a analizar detalladamente qué es lo que hace el sistema representado en la figura siguiente.



Un computador reversible que no genera entropía. La bifurcación, BIF, es en realidad un proceso de copia.

En la figura podemos identificar un conjunto de elementos importantes que nos van a servir para configurar un computador reversible de propósito general, y además limpito, ordenadito y aseado. También podemos reconocer tres fases distintas, que utilizaremos para tratar de simplificar la descripción del proceso de la computación reversible. Los elementos importantes de la arquitectura propuesta son los siguientes:

1. Registro |x| que llamaremos registro de copia
2. Registro |y| que llamaremos registro de datos
3. Registro |z| que llamaremos registro de control

4. Unidad ⟨M⟩ que es el sistema lógico reversible que realiza la computación

5. Unidad ⟨BIF⟩ que realiza una operación de bifurcación ⟨FO⟩

6. Unidad ⟨W⟩ que invierte las operaciones del sistema lógico ⟨M⟩

Fase 1

- El registro de copia, inicialmente en el estado |x1⟩ = |0⟩ pasa directamente a la siguiente fase como |x2⟩ = |0⟩.

- El registro de datos, inicialmente en el estado |y1⟩ = |Datos⟩, es procesado por la unidad ⟨M⟩, que realiza la computación reversible, y pasa a la siguiente fase como |y2⟩ = |Resultado⟩.

- El registro de control, inicialmente en el estado |z1⟩ = |0⟩ es introducido también en la unidad ⟨M⟩ para controlar el proceso de la computación, y sale completamente desordenado, y convertido en un conjunto de bits basura que, sin embargo, aseguran la reversibilidad. Este registro pasa a la siguiente fase como|z2⟩ = |Basura⟩

Fase 2

- El registro de copia recibe la información procedente de una bifurcación realizada sobre |y2⟩, por lo que pasa a la siguiente fase como |x3⟩ = |Resultado⟩.

- El registro |y2⟩ = |Resultado⟩ sufre una bifurcación, y pasa a la siguiente fase como |y3⟩ = |Resultado⟩

- El registro de control no sufre ninguna operación, y pasa directamente a la siguiente fase como |z3⟩ = |Basura⟩.

Fase 3

- El registro de copia ya no se modifica. Su estado final es $|x^*\rangle=|\text{Resultado}\rangle$
- El registro $|y3\rangle = |\text{Resultado}\rangle$ es ahora procesado por la unidad $\langle W\rangle$, que deshace en orden inverso las operaciones realizadas previamente en $\langle M\rangle$, por lo que se reconstruye el registro de datos. Su estado final es $|y^*\rangle = |\text{Datos}\rangle$.
- El registro $|z3\rangle = |\text{Basura}\rangle$ es ahora procesado por la unidad $\langle W\rangle$ que deshace en orden inverso las operaciones realizadas en $\langle M\rangle$, por lo que se reconstruye el registro de ceros. Su estado final es $|z^*\rangle = |0\rangle$.

El resultado final de la computación, que por supuesto es absolutamente reversible, incluye un registro con los datos de partida, un registro con el resultado de la computación, y un registro con los ceros de control. Todo perfectamente ordenadito, y sin ninguna información basura. Eso sí, a expensas de sacrificar un registro extra, inicialmente lleno de ceros,… nada grave.

Nótese que $\langle M\rangle$ es un sistema lógico reversible capaz de realizar computaciones todo lo complicadas que queramos. $\langle W\rangle$ deshace en orden inverso todas las operaciones previamente realizadas por $\langle M\rangle$, y también es reversible. Por otra parte, lo que nosotros hemos señalado como operación de bifurcación $\langle FO\rangle$ es en realidad un proceso de copia.

Pero no todo lo que hemos dicho es estrictamente cierto… en realidad parece que con la arquitectura propuesta no se genera ningún desorden, al menos desde una perspectiva global. Sin embargo no debemos olvidar que estamos trabajando con máquinas reversibles, por lo que los bits pueden ir de acá para allá. Por lo tanto, para conducir la computación en un sentido determinado, y evitar que

los bits se comporten como bolas saltarinas, hay que darle un empujón al sistema, y es aquí cuando se genera algún desorden. También es cierto que pasan cosas increíbles cuando reiniciamos nuestra computadora y la preparamos para una nueva operación. Pero antes nos detendremos en tratar de averiguar qué podría pasar cuando los componentes de nuestras computadoras sean lo suficientemente pequeños como para que los fenómenos cuánticos empiecen a tener su importancia.

--..--

INFORMACIÓN CUÁNTICA

En el modelo clásico de computación el bit es la unidad básica de información. Un bit puede tener dos valores distintos que se denotan 0 y 1 respectivamente. Desde un punto de vista un poco más formal, un bit es un elemento del conjunto V = {0,1}. Una cadena de n bits se puede considerar como un elemento del producto cartesiano:

$$V^n = V \times \cdots \times V$$

Una cadena de bits puede representar cualquier información. Para ello basta establecer un mecanismo de codificación. Por otra parte, en el modelo clásico de computación, un algoritmo es un mecanismo para manipular cadenas de bits, y desde el punto de vista formal se puede considerar como un mecanismo para evaluar funciones booleanas. Así, dada una cadena de n bits, α, el algoritmo la modifica generando otra cadena de n bits, β.

Si llamamos f a la función booleana de $V^n \rightarrow V^n$ / $f(\alpha) = \beta$ entonces el algoritmo es un mecanismo para evaluar f. Desde una perspectiva funcional, en un algoritmo hay que detallar el mecanismo de manipulación de la cadena de bits hasta reducirlo a una secuencia de puertas lógicas, ya que las computadoras clásicas sólo son capaces de evaluar puertas lógicas, pero no son capaces de evaluar funciones booleanas genéricas. En este sentido, recordaremos que las puertas lógicas {not}, {or} y {and} permiten definir cualquier función booleana.

Formalización del Qubit

En el modelo cuántico de computación la unidad de información básica es el qubit o bit cuántico. Un qubit puede estar en dos estados distintos que se denotan $|0\rangle$ y $|1\rangle$ respectivamente. Físicamente se representa por un sistema cuántico de dos estados. El sistema cuántico de dos estados más conocidos es el spin de un electrón. En un sistema de este tipo podemos representar el spin -(1/2) por el estado $|0\rangle$ y el spin +(1/2) por el estado $|1\rangle$. El qubit es un elemento del espacio de

Hilbert de funciones de onda más simple no trivial de dos dimensiones, generado por los kets {|0⟩ , |1⟩}, elementos de la base, y que convencionalmente pueden elegirse en una representación particular como:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Estos dos vectores son ortonormales, lo cual significa que bajo el producto escalar ⟨x|y⟩ definido en el espacio, los vectores base se comportan de la siguiente forma:

$$\langle 0|0\rangle = \langle 1|1\rangle = 1 \qquad : \qquad \langle 0|1\rangle = \langle 1|0\rangle = 0$$

En las dos últimas ecuaciones los vectores bra ⟨0|, ⟨1|, duales de los ket |0⟩, |1⟩, se obtienen como los traspuestos hermíticos de los ket y se representan de la siguiente manera:

$$\langle 0| = (1\ 0)$$

$$\langle 1| = (0\ 1)$$

Un qubit, en general, se presenta como una superposición o combinación lineal de los estados básicos |0⟩ y |1⟩ tal que:

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

donde las amplitudes de probabilidad α y β son en general números complejos, esto es, contienen información de fase. Como en cualquier medida en mecánica cuántica, los cuadrados de estos coeficientes determinan respectivamente la probabilidad de obtener en una medida los resultados |0⟩ y |1⟩ . Puesto que la probabilidad total tiene que ser la unidad, α y β se deben relacionar por la ecuación:

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

Esta ecuación simplemente asegura que en la medición se obtiene un estado o el otro. Debido a su naturaleza cuántica, cualquier medida del qubit altera

inevitablemente su estado, se rompe la superposición y colapsa en aquel estado de base que ha resultado de la medida, y {α, β} se transforma irreversiblemente en {0, 1}. Volveremos más adelante sobre esta cuestión.

Alternativamente, el qubit también puede describirse por medio de una matriz densidad. Para un qubit en el estado $|\psi\rangle$ el operador proyección correspondiente es:

$$\rho_\psi = |\psi\rangle \langle\psi|$$

En contraste con el vector de estado, la matriz de densidad está definida de forma unívoca. Mediante matrices densidad, es posible describir qubits cuyo estado no es bien conocido, los llamados «estados mezcla». En general se puede escribir la matriz densidad de un qubit en la forma:

$$\rho = \frac{1}{2}\left(\mathbf{1} + \sum_{\mathbf{i=1}}^{3} c_i \sigma_i\right) \qquad \text{tal que} \qquad c_1^2 + c_2^2 + c_3^2 \leq 1$$

donde **1** es la matriz unidad 2 × 2 y σ_i son las matrices de Pauli, que en su representación lineal más común, y para el caso de espín 1/2 tienen la siguiente forma:

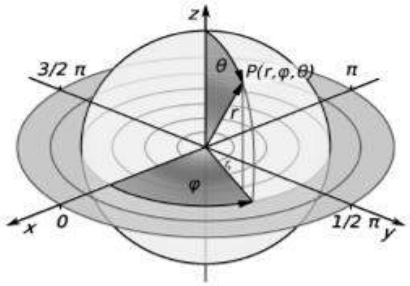
$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \qquad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \qquad \sigma_x = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

y la probabilidad de encontrar el estado $|\psi\rangle$ en una medida viene dada por:

$$P_\psi = \langle\psi|\rho||\psi\rangle$$

El espacio de estados del qubit se puede representar mediante un espacio vectorial complejo bidimensional de módulo 1. Equivalentemente, se pueden representar puntos en la superficie de una esfera; esta superficie se llama esfera de Bloch. Cada estado del qubit corresponde a un punto de la superficie de una esfera de radio unidad. Esto esencialmente significa que un qubit tiene dos grados de libertad locales. Estos grados de libertad podrían ser la longitud y

latitud, o como es más habitual, dos ángulos θ y ϕ en coordenadas esféricas, como se muestra en la figura.

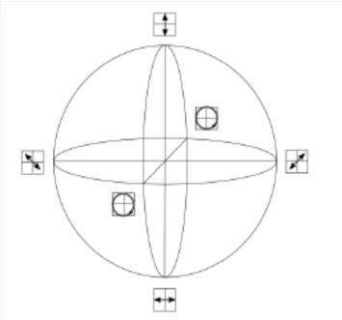


Representación esquemática de coordenadas esféricas.

Si se asigna el estado $|1\rangle$ al «polo norte» de la esfera, el estado correspondiente es:

$$|\psi\rangle = \sin\left(\frac{\theta}{2}\right)\exp\left(\frac{-i\phi}{2}\right)|0\rangle + \cos\left(\frac{\theta}{2}\right)\exp\left(\frac{i\phi}{2}\right)|1\rangle$$

Un caso intuitivo para el uso de la esfera de Bloch es el de la partícula de espín 1/2, en el que el punto sobre la esfera indica la dirección en la que el qubit es función propia de la proyección del espín, esto es, donde se va a obtener un valor determinado, no probabilístico, para S_z que, sin embargo, es aplicable a cualquier qubit. En la siguiente figura, a modo de ejemplo, se representan algunos estados de un qubit basado en la polarización de un fotón.



Polarización de un fotón.

En la figura puede comprobarse que los estados $|0\rangle$ y $|1\rangle$ son equivalentes a la polarización vertical y horizontal, dos de las combinaciones lineales con el mismo peso de $|0\rangle$ y $|1\rangle$ son las polarizaciones diagonales, y las otras dos son las polarizaciones circulares.

Sistemas de Qubits

El 1-qubit normalizado más general que se puede formar en este espacio es la superposición lineal de los dos elementos de la base, es decir:

$$|x\rangle = a_0 |0\rangle + a_1 |1\rangle : a_0 , a_1 \in C : |a_0|^2 + |a_1|^2 = 1$$

Un bit tiene dos valores posibles y un qubit puede estar en dos estados posibles. Sin embargo un qubit puede estar además en estados intermedios, es decir, en estados que son combinación lineal de los estados $|0\rangle$ y $|1\rangle$. Esta es la primera gran diferencia entre los modelos de computación clásico y cuántico. Por ejemplo, el spin de un electrón puede estar en estado:

$$\psi = (1/2) |0\rangle + (\sqrt{3}/2) |1\rangle$$

La primera conclusión importante es que un qubit es un vector de un espacio vectorial generado por los dos estados, es decir, es un vector de:

$$V = L(|0\rangle, |1\rangle)$$

y, según la mecánica cuántica, V es un espacio de Hilbert complejo en el que

$$B =[|0\rangle, |1\rangle]$$

es una base ortonormal y los estados son vectores unitarios. Entonces un qubit puede estar en cualquier estado:

$$|x\rangle = a_0 |0\rangle + a_1 |1\rangle : a_0 , a_1 \in C : |a_0|^2 + |a_1|^2 = 1$$

Los coeficientes se denominan amplitudes. Resulta relativamente fácil entender los estados de la base pero no sucede lo mismo con los estados

intermedios. Volvamos a retomar el ejemplo del spin de un electrón y analicemos un estado intermedio. En este caso el qubit ψ no tiene spin definido. Para convencerse de ello basta recordar que el spin de un electrón es una magnitud física que está cuantificada. Por lo tanto el estado intermedio ψ no tiene spin definido.

La información que contiene un qubit es evidentemente muy pequeña. Para poder representar cantidades mayores de información se recurre a sistemas de n-qubits. Si queremos describir el estado de un sistema que consiste no sólo de un qubit sino de un conjunto de n qubits cuánticos tenemos que realizar el producto tensorial de los n qubits individuales.

Para empezar con un ejemplo sencillo supongamos que n = 2, por ejemplo el spin de un sistema de dos electrones. El spin de cada electrón puede estar en dos estados que combinados generan cuatro estados para el sistema de 2-qubits. Estos estados se denotan $|00\rangle$, $|01\rangle$, $|10\rangle$ y $|11\rangle$ respectivamente. Para n = 2, es decir, dos qubits o un 2-qubit, la dimensión del espacio es $2^2 = 4$ y la base de este espacio es:

$$|u_i\rangle \otimes |v_j\rangle = |u_i , v_j\rangle$$

En esta última expresión: i, j = 0, 1, entonces:

$$|u_0\rangle = |0\rangle : |u_1\rangle = |1\rangle : |v_0\rangle = |0\rangle : |v_1\rangle = |1\rangle$$

Las relaciones expresadas por las ecuaciones anteriores permiten definir la base del espacio estado cuatro-dimensional como:

$$\{|0\rangle \otimes |0\rangle , |0\rangle \otimes |1\rangle , |1\rangle \otimes |0\rangle , |1\rangle \otimes |1\rangle\}$$

lo que puede ser escrito en una forma completamente equivalente como:

$$\{|00\rangle , |01\rangle , |10\rangle , |11\rangle\}$$

Igual que ocurre con los qubits, un 2-qubit puede estar en un estado intermedio, es decir, puede ser una combinación lineal de los cuatro estados anteriores. Por ejemplo un 2-qubit puede estar en el estado:

$$\psi = (1/4)\,|00\rangle + (\sqrt{3}/4)\,|01\rangle + (\sqrt{3}/4)\,|10\rangle + (3/4)\,|11\rangle$$

Entonces un 2-qubit es un vector del espacio vectorial:

$$V2 = L(|00\rangle, |01\rangle, |10\rangle, |11\rangle)$$

que, según la mecánica cuántica, es un espacio de Hilbert complejo en el que:

$$B2 = [|00\rangle, |01\rangle, |10\rangle, |11\rangle]$$

es una base ortonormal y los estados son vectores unitarios. De este modo, como ya hemos dicho, un 2-qubit puede estar en cualquier estado de la forma:

$$\Psi = a|00\rangle + b|01\rangle + c|10\rangle + d|11\rangle : a, b, c, d \in C : |a|^2 + |b|^2 + |c|^2 + |d|^2 = 1$$

La descripción que hemos hecho de los sistemas de 2-qubits es completa. Sin embargo no sabemos qué relación existe entre el espacio vectorial V^2 y los espacios vectoriales V asociados a los dos qubits, considerados como sistemas independientes. Pero V^2 es el producto tensorial $V \otimes V$, por lo tanto los vectores de la base B^2 corresponden a los distintos productos tensoriales de los vectores de B:

$$|00\rangle = |0\rangle \otimes |0\rangle : |01\rangle = |0\rangle \otimes |1\rangle : |10\rangle = |1\rangle \otimes |0\rangle : |11\rangle = |1\rangle \otimes |1\rangle$$

El estado de un 2-qubit se denomina estado entrelazado si no se puede describir en términos de los estados de los qubits que componen el sistema. Desde un punto de vista un poco más formal esto significa que no puede ponerse como el producto tensorial de dos estados de un solo qubit. Por ejemplo, el 2-qubit de la expresión :

$$\psi = (1/4)\,|00\rangle + (\sqrt{3}/4)\,|01\rangle + (\sqrt{3}/4)\,|10\rangle + (3/4)\,|11\rangle$$

no está en un estado entrelazado pues se puede escribir como un producto tensorial:

$$\psi = \left(\frac{1}{2}|0\rangle + \frac{\sqrt{3}}{2}|1\rangle\right) \otimes \left(\frac{1}{2}|0\rangle + \frac{\sqrt{3}}{2}|1\rangle\right)$$

Efectivamente: Sea $\psi = \psi_1 \otimes \psi_2$ / $\psi_1 = a\,|0\rangle + b\,|1\rangle$ y $\psi_2 = c\,|0\rangle + d\,|1\rangle$ en los que se tienen que cumplir las condiciones de normalización:

$$a^2 + b^2 = 1 \qquad y \qquad c^2 + d^2 = 1$$

Por otra parte "ac = 1/4 ; ad = √3/4 ; bc = √3/4 ; bd = 3/4". Y este sistema de ecuaciones tiene solución, por ejemplo:

$$a = 1/2 : b = \sqrt{3}/2 : c = 1/2 : d = \sqrt{3}/2$$

Sin embargo se puede probar fácilmente que el estado de la siguiente expresión sí está entrelazado:

$$\psi = (1/\sqrt{3})\,|00\rangle + (1/\sqrt{3})\,|01\rangle + (1/\sqrt{3})\,|10\rangle$$

En realidad casi todos los estados están entrelazados. Si elegimos aleatoriamente los coeficientes a, b, c y d (un punto sobre la esfera de radio 1 centrada en el origen en R^8) la probabilidad de que el resultado sea un estado entrelazado es 1.

Si recordamos que el 2-qubit normalizado más general "$|x_1\,x_2\rangle = |x_1\rangle \otimes |x_2\rangle$" que se puede formar en este espacio es la superposición lineal de los cuatro elementos de la base, es decir:

$$|x_1\,x_2\rangle = a_0\,|00\rangle + a_1\,|01\rangle + a_2\,|10\rangle + a_3\,|11\rangle \text{ tal que } a_0, a_1, a_2, a_3 \in C$$

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

entonces la generalización de 2-qubits a n-qubits resulta ahora muy sencilla. Un n-qubit es un vector unitario del espacio de Hilbert complejo:

$$Vn = V \otimes \cdots \otimes V$$

en el que:

$$Bn = [|0 \cdots 00\rangle, |0 \cdots 01\rangle, |0 \cdots 10\rangle, ..., |1 \cdots 11\rangle]$$

es una base ortonormal, llamada base computacional. Un vector genérico de la base Bn se puede ver como un producto tensorial del siguiente modo:

$$|x_1\, x_2 \ldots x_n\rangle = |x_1\rangle \otimes |x_2\rangle \otimes \ldots \otimes |x_n\rangle \; ; \quad \text{con} \quad x_1, x_2, \ldots, x_n \in \{0, 1\}$$

Entonces, el n-qubit normalizado más general

$$|x_1 \ldots x_n\rangle = |x_1\rangle \otimes \ldots \otimes |x_n\rangle$$

viene dado por la superposición lineal de los 2^n elementos de la base:

$$\left|x_1, x_2, ..., x_n\right\rangle = \sum_{i=0}^{2^n-1} a_i \left|i\right\rangle \quad : \quad a_0, ..., a_{n-1} \in C \quad : \quad \sum_{i=0}^{2^n-1} \left|a_i\right|^2 = 1$$

Dicho de otro modo: el estado conjunto de un sistema formado por "n" qubits se describe como un punto en el espacio de Hilbert de dimensión 2^n, que implica el producto tensorial de los "n" espacios de Hilbert de cada qubit.

Se puede representar el estado compuesto de forma compacta, por ejemplo:

$$|0100\rangle = |0\rangle_1 \otimes |1\rangle_2 \otimes |0\rangle_3 \otimes |0\rangle_4$$

donde la posición o el índice {1-4} indican el qubit y el valor {0,1} indican el estado de cada qubit. Todo producto directo entre estados de qubits da lugar a un estado conjunto de n-qubits, por ejemplo:

$$(1/\sqrt{2})\,(|0\rangle_1 + |1\rangle_1) \otimes (1/\sqrt{2})\,(|0\rangle_2 - |1\rangle_2) = (1/2)\,(|00\rangle - |01\rangle + |10\rangle - |11\rangle)$$

En cambio no se aplica lo contrario, y existen estados conjuntos de n-qubits que no se pueden describir como producto de los estados individuales de los "n" qubits, por ejemplo:

$$[1/\sqrt{2}]\ \{|00\rangle + |11\rangle\}$$

Estos estados se conocen como entrelazados porque los estados de los dos qubits no son independientes.

Una convención utilizada en computación cuántica es:

$$|\ x_1\ x_2\ \ldots\ x_m\ \rangle \equiv |x\rangle$$

donde x_1 , x_2 , . . . , x_m es la representación binaria del entero "x":

$$x = x_1 2^{m-1} + x_2 2^{m-2} + \ldots + x_{m-1} 2^1 + x_m 2^0$$

Según esto, la base de un espacio formado por "n" qubits cuya dimensión es 2^n está formada por:

$$\{|0\rangle\ ,\ |1\rangle\ ,\ |2\rangle\ ,\ \ldots\ ,\ |2^n - 1\rangle\}$$

La cadena de bits "$x_1\ x_2\ \ldots\ x_n$" la podemos interpretar como un número natural "x" representado en el sistema de numeración binario. De este modo los vectores de la base Bn se identifican con los números naturales x que cumplen $0 \leq x < 2^n$ (números con n dígitos binarios).

Una vez identificada la cadena de bits "x1 x2 . . . xn" con el número natural "x", se puede escribir "x" en el sistema de numeración decimal. En definitiva podemos escribir Bn =[$|0\rangle$, $|1\rangle$, $|2\rangle$, . . . , $|2^n - 1\rangle$]

La identificación de los vectores de la base con cadenas de "n" bits es importante para codificar información en un n-qubit, mientras que identificarlos con números naturales tiene que ver con nuestra predilección por el sistema de numeración decimal. Con esta notación un n-qubit se puede escribir del siguiente modo:

$$\psi = \sum_{x=0}^{2^n-1} a_x |x\rangle \text{ tal que } \sum_{x=0}^{2^n-1} |a_x|^2 = 1$$

Conviene resaltar que la dimensión del espacio es exponencial, concretamente 2^n. Esta es la propiedad clave del denominado paralelismo cuántico y de la enorme capacidad de un n-qubit para almacenar información. Por ejemplo, el par de números (131, 211) se puede codificar en una cadena de 16 bits, 8 para cada número, que se puede representar mediante el 16-qubit:

$$\psi_1 = |1000001111010011\rangle = |27641\rangle$$

Sin embargo en un 16-qubit se puede codificar mucha más información. Así el estado:

$$\psi_2 = \frac{1}{256}\sum_{x=0}^{65535}|x\rangle$$

es una combinación lineal de todos los pares de números de 8 dígitos binarios desde el (0, 0) hasta el (255, 255) ambos incluidos. En este ejemplo anterior los ocho primeros qubits codifican el primer número del par mientras los ocho qubits restantes codifican el segundo número.

Para facilitar la codificación de información se pueden agrupar los qubits en registros. Formalmente un registro de tamaño k es un conjunto consecutivo de k qubits. Se puede denotar por "|x⟩, |y⟩, |z⟩. . ." donde los números "x, y, z . . ." son números con "k" dígitos binarios. En el ejemplo anterior, llamando |x⟩ al registro completo de 16 qubits, |y⟩ al registro de los 8 primeros qubits y |z⟩ al registro de los 8 últimos qubits, los estados ψ_1 y ψ_2 se pueden escribir del siguiente modo:

$$\psi_1 = |10000011\rangle \otimes |11010011\rangle = |131\rangle \otimes |211\rangle$$

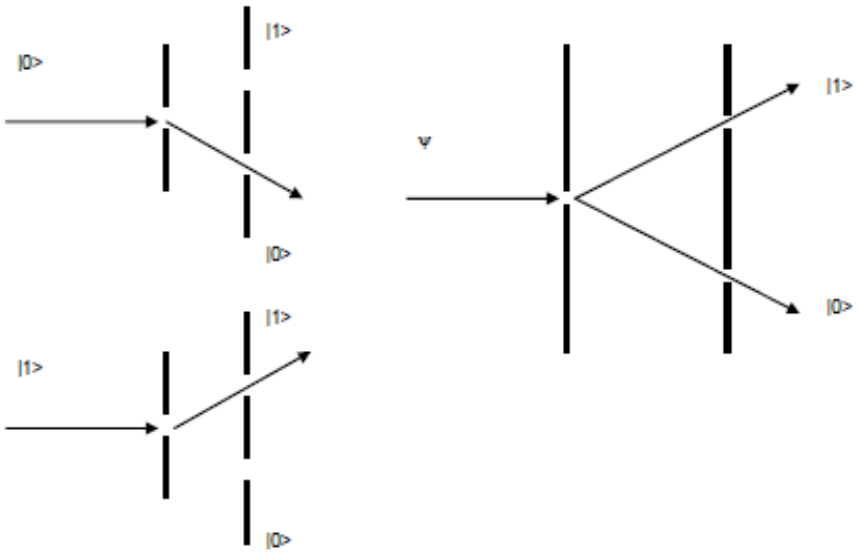
$$\psi_2 = \frac{1}{256}\sum_{y=0}^{255}\sum_{z=0}^{255}|y\rangle \otimes |z\rangle$$

Naturaleza Probabilística de la Medida de Qubits

Siempre es posible medir el valor de un bit pero generalmente no es posible medir el estado de un qubit. Vamos a estudiar lo que ocurre al medir el qubit definido por la expresión:

$$\psi = (1/2)\,|0\rangle + (\sqrt{3}/2)\,|1\rangle$$

Para ello empleamos el dispositivo esquematizado en la figura. El proceso de medida consiste en hacer pasar al electrón a través de la rendija del panel 1. Cuando pasa por la rendija el electrón atraviesa un campo magnético que desvía su trayectoria según el valor de su spin, hasta que finalmente el electrón atraviesa una de las dos rendijas del panel.



Procedimiento de medida de los qubits $|0\rangle$, $|1\rangle$ y $|\psi\rangle$. *El qubit* $|\psi\rangle$ *no tiene spin definido, y por ello tiene cierta probabilidad de salir por arriba o por abajo.*

El qubit ψ que estamos analizando es un estado intermedio y en consecuencia no tiene spin definido. El electrón tiene posibilidad de desviarse tanto hacia abajo como hacia arriba. En primer lugar conviene aclarar que el electrón saldrá por una de las dos rendijas del panel 2. Si pudiese alcanzar

posiciones intermedias entre las rendijas del panel 2 el spin del electrón no estaría cuantificado.

Una vez asumido este hecho vamos a justificar por qué tiene posibilidad de salir tanto por la rendija inferior como por la rendija superior. Si sólo tuviese posibilidad de salir por una de las rendijas, supongamos que por la superior, significaría que se trata del estado $|1\rangle$ pues tendría spin definido. Pero si el electrón ha pasado por la rendija inferior su spin, después de la medida, sólo puede ser -1/2, y su estado $|0\rangle$. De modo análogo, si el electrón ha pasado por la rendija superior su spin, después de la medida, sólo puede ser +1/2 y su estado $|1\rangle$. El proceso de medida, además de dar una información incompleta sobre el qubit, lo modifica. De alguna manera, el proceso de medida "obliga al qubit a decidirse" por uno de los dos estados de la base.

Una vez hecho el análisis cualitativo de la medida del qubit es conveniente describir cuantitativamente el proceso. Los postulados de la mecánica cuántica establecen que la probabilidad p_0 ó p_1 de que el estado final del qubit sea $|0\rangle$ ó $|1\rangle$ es igual al cuadrado del módulo de la amplitud de $|0\rangle$ o de $|1\rangle$) en la combinación lineal. Para el qubit Ψ del ejemplo el resultado final será $|0\rangle$ con probabilidad $p_0=1/4$ y $|1\rangle$ con probabilidad $p_1=3/4$

En la tabla siguiente se resume el proceso de medida de un qubit. Diremos que el resultado de la medida es 0 ó1 si el estado final es $|0\rangle$ ó $|1\rangle$.

Estado	Medida	Estado final	Probabilidad
$a\|0\rangle + b\|1\rangle$	0	$\frac{a}{\|a\|}\|0\rangle$	$p_0 = \|a\|^2$
$a\|0\rangle + b\|1\rangle$	1	$\frac{b}{\|b\|}\|1\rangle$	$p_1 = \|b\|^2$

Tabla resumen del proceso de medida de un qubit.

En un 2-qubit podemos medir el primer qubit o el segundo qubit. El proceso en ambos casos es similar. Supongamos pues que vamos a medir el primer qubit. Tomemos como ejemplo el estado:

$$\psi = (1/\sqrt{3})\,|00\rangle + (1/\sqrt{3})\,|01\rangle + (1/\sqrt{3})\,|10\rangle$$

Después de la medida, el primer qubit debe estar en estado $|0\rangle$ o en estado $|1\rangle$. Por lo tanto el 2-qubit deberá estar, después de la medida, en uno cualquiera de los siguientes estados:

$$|0\rangle \otimes [a\,|0\rangle + b\,|1\rangle] = a\,|00\rangle + b\,|01\rangle : \; a, b \in C \text{ tal que } |a|^2 + |b|^2 = 1$$

$$|1\rangle \otimes [a\,|0\rangle + b\,|1\rangle] = a\,|10\rangle + b\,|11\rangle : \; a, b \in C \text{ tal que } |a|^2 + |b|^2 = 1$$

Para obtener el estado resultante del proceso de medida escribimos el estado ψ como una combinación lineal de dos estados, ψ_0 y ψ_1, en los que el primer qubit está en estado $|0\rangle$ y $|1\rangle$ respectivamente. Obtenemos las expresiones correspondientes aplicando las restricciones de los qubits:

$$\psi = \alpha\,\psi_0 + \beta\,\psi_1 \;/\; |\alpha|^2 + |\beta|^2 = 1$$

$$\psi_0 = x_{00}\,|00\rangle + x_{01}\,|01\rangle \;/\; |x_{00}|^2 + |x_{01}|^2 = 1$$

$$\psi_1 = x_{10}\,|10\rangle + x_{11}\,|11\rangle \;/\; |x_{10}|^2 + |x_{11}|^2 = 1$$

$$x_{11} = 0 \text{ (el estado } |11\rangle \text{ no está en } \psi) \rightarrow x_{10} = 1 \rightarrow \beta = 1/\sqrt{3}$$

$$|\alpha|^2 + |\beta|^2 = 1 \rightarrow \alpha^2 = 1 - (1/3) = 2/3 \rightarrow \alpha = \sqrt{2}/\sqrt{3}$$

$$(\sqrt{2}/\sqrt{3})\,x_{00} = 1/\sqrt{3} \;;\; (\sqrt{2}/\sqrt{3})\,x_{01} = 1/\sqrt{3} \rightarrow x_{00} = x_{01} = 1/\sqrt{2}$$

Por lo tanto:

$$\psi = (\sqrt{2}/\sqrt{3})\,\psi_0 + (1/\sqrt{3})\,\psi_1$$

$$\psi_0 = (1/\sqrt{2})\ |00\rangle + (1/\sqrt{2})\ |01\rangle : \psi_1 = |10\rangle$$

A tenor de las expresiones anteriores si el resultado de la medida es 0 el estado final será ψ_0 y si es 1 el estado final será ψ_1. Además la probabilidad p_0 o p_1 de que la medida sea 0 ó 1 es igual al módulo al cuadrado del coeficiente de ψ_0 o ψ_1. En tabla se resume el proceso de medida de este caso particular.

Estado	Medida	Estado final	Probabilidad
$\frac{1}{\sqrt{3}}\|00\rangle + \frac{1}{\sqrt{3}}\|01\rangle + \frac{1}{\sqrt{3}}\|10\rangle$	0	$\Psi_0 = \frac{1}{\sqrt{2}}\|00\rangle + \frac{1}{\sqrt{2}}\|01\rangle$	$p_0 = \frac{2}{3}$
$\frac{1}{\sqrt{3}}\|00\rangle + \frac{1}{\sqrt{3}}\|01\rangle + \frac{1}{\sqrt{3}}\|10\rangle$	1	$\Psi_1 = \|10\rangle$	$p_1 = \frac{1}{3}$

Proceso de medida del qubit: $\psi = (1/\sqrt{3})\ |00\rangle + (1/\sqrt{3})\ |01\rangle + (1/\sqrt{3})\ |10\rangle$

La probabilidad p_0 y el estado fnal ψ_0 tienen otra interpretación: ψ_0 es la proyección ortogonal normalizada de ψ sobre el subespacio L($|00\rangle$, $|01\rangle$) y p_0 es la norma al cuadrado de dicha proyección, es decir, la suma de los módulos al cuadrado de las amplitudes de $|00\rangle$ y $|01\rangle$ en el estado ψ. Obviamente las interpretaciones de p_1 y ψ_1 son análogas.

Veamos algunos ejemplos:

- Sea $|x\rangle$ un qubit normalizado. Si se realiza una medida sobre la base $\{|u_0\rangle , |u_1\rangle\}$, la probabilidad de encontrar el qubit en el estado $|u_i\rangle$, denotada por P ($|u_i\rangle$), está dada por:

$$P\ (|u_i\rangle) = |\langle u_i \mid x\rangle\ |^2$$

- La medición de un qubit $|x\rangle = a_0\ |0\rangle + a_1\ |1\rangle$ sobre la base $\{|0\rangle , |1\rangle\}$ genera las siguientes probabilidades:

$$P(|0\rangle) = |\langle 0|x\rangle^2 = |\langle 0|(a_0|0\rangle + a_1|1\rangle)|^2 = |a_0\langle 0|0\rangle + a_1\langle 0|1\rangle|^2 = |a_0|^2$$

$$P(|1\rangle) = |\langle 1|x\rangle^2 = |\langle 1|(a_0|0\rangle + a_1|1\rangle)|^2 = |a_0\langle 1|0\rangle + a_1\langle 1|1\rangle|^2 = |a_1|^2$$

- La medición de un qubit $|x\rangle = a_0 |0\rangle + a_1 |1\rangle$ sobre la base

  $\{|0\rangle + |1\rangle , |0\rangle - |1\rangle\}$ genera las siguientes probabilidades:

$$P(|0\rangle + |1\rangle) = |(\langle 0| + \langle 1|)|x\rangle|^2 = |(\langle 0| + \langle 1|)(a_0|0\rangle + a_1|1\rangle)|^2 =$$

$$= |a_0\langle 0|0\rangle + a_1\langle 0|1\rangle + a_0\langle 1|0\rangle + a_1\langle 1|1\rangle|^2 = |a_0 + a_1|^2$$

$$P(|0\rangle - |1\rangle) = |(\langle 0| - \langle 1|)|x\rangle|^2 = |(\langle 0| - \langle 1|)(a_0|0\rangle + a_1|1\rangle)|^2 =$$

$$= |a_0\langle 0|0\rangle + a_1\langle 0|1\rangle - a_0\langle 1|0\rangle - a_1\langle 1|1\rangle|^2 = |a_0 - a_1|^2$$

La medida de los qubits de un 2-qubit:

$$|x_1 x_2\rangle = a_0 |00\rangle + a_1 |01\rangle + a_2 |10\rangle + a_3 |11\rangle$$

sobre la base $\{|0\rangle , |1\rangle\}$ genera las siguientes probabilidades:

- Probabilidad de encontrar el primer qubit en el estado $|0\rangle$, $P_1 (|0\rangle)$
- Probabilidad de encontrar el primer qubit en el estado $|1\rangle$, $P_1 (|1\rangle)$
- Probabilidad de encontrar el segundo qubit en el estado $|0\rangle$, $P_2 (|0\rangle)$
- Probabilidad de encontrar el segundo qubit en el estado $|1\rangle$, $P_2 (|1\rangle)$

Estas probabilidades están dadas por:

$$P_1(|0\rangle) = |a_0|^2 + |a_1|^2 \quad : \quad P_1(|1\rangle) = |a_2|^2 + |a_3|^2$$

$$P_2(|0\rangle) = |a_0|^2 + |a_2|^2 \quad : \quad P_2(|1\rangle) = |a_1|^2 + |a_3|^2$$

Si una vez realizada la medida sobre el primer qubit del 2-qubit

$$|x_1 x_2\rangle = a_0 |00\rangle + a_1 |01\rangle + a_2 |10\rangle + a_3 |11\rangle$$

se obtiene que está en el estado $|0\rangle$, el 2-qubit evoluciona a un nuevo estado normalizado dado por:

$$|x'\,y'\rangle = \frac{a_0|00\rangle + a_1|01\rangle}{\sqrt{|a_0|^2 + |a_1|^2}}$$

y si se obtiene que está en el estado |1⟩, el 2-qubit evoluciona a un nuevo estado normalizado dado por:

$$|x'\,y'\rangle = \frac{a_2|10\rangle + a_3|11\rangle}{\sqrt{|a_2|^2 + |a_3|^2}}$$

Expresiones similares a las anteriores se obtienen para los resultados de la medida del segundo qubit del 2-qubit.

En un sistema de n-qubits podemos medir cualquiera de los qubits, por ejemplo el k-ésimo. El proceso es análogo al que ya hemos explicado para 2-qubits y está resumido en la tabla.

Estado	Medida	Estado final	Probabilidad
$\sum_{x=0}^{2^n-1} a_x\|x\rangle$	0	$\Psi_0 = \frac{1}{\sqrt{p_0}} \sum_{\substack{0\le x<2^n \\ x_k=0}} a_x\|x\rangle$	$p_0 = \sum_{\substack{0\le x<2^n \\ x_k=0}} \|a_x\|^2$
$\sum_{x=0}^{2^n-1} a_x\|x\rangle$	1	$\Psi_1 = \frac{1}{\sqrt{p_1}} \sum_{\substack{0\le x<2^n \\ x_k=1}} a_x\|x\rangle$	$p_1 = \sum_{\substack{0\le x<2^n \\ x_k=1}} \|a_x\|^2$

Proceso de medida de un sistema de n-qubits.

Algunos Ejemplos de Medidas de Qubits

Ya hemos visto que una de las características particulares y no intuitivas de los sistemas cuánticos es la relacionada con la existencia de estados enredados o entrelazados. La existencia de estos estados cuánticos permiten afirmar que la

descripción del estado de un sistema cuántico no puede ser siempre realizada mediante la descripción de los elementos que lo componen.

Un estado cuántico de "n" qubits se dice que está enredado si no puede ser expresado como el producto tensorial de los estados de cada uno de los "n" qubits que lo componen. Sean dos qubits $|x\rangle$ y $|y\rangle$ descritos por las expresiones:

$$|x\rangle = a\,|0\rangle + b\,|1\rangle \;:\; |y\rangle = c\,|0\rangle + d\,|1\rangle \quad \text{tal que:} \quad a, b, c, d \in C$$

El producto tensorial de estos dos qubits nos genera el 2-qubit:

$$|x\rangle \otimes |y\rangle = |x, y\rangle = (a\,|0\rangle + b\,|1\rangle) \otimes (c\,|0\rangle + d\,|1\rangle =$$

$$= ac\,|0, 0\rangle + ad\,|0, 1\rangle + bc\,|1, 0\rangle + bd\,|1, 1\rangle$$

Un estado de dos qubits es un estado enredado si no puede ser expresado en la forma de la ecuación anterior. Así, el estado:

$$|\psi\rangle = 1/2\,(|0, 0\rangle - |0, 1\rangle + |1, 0\rangle - |1, 1\rangle)$$

es un estado no enredado porque el sistema de ecuaciones:

$$ac = 1/2 : ad = -1/2 : bc = 1/2 : bd = -1/2$$

tiene la solución: $a = b = c = 1/\sqrt{2}$; $d = -1/\sqrt{2}$, y por lo tanto:

$$|\psi\rangle = \frac{1}{2}\left(|0,0\rangle - |0,1\rangle + |1,0\rangle - |1,1\rangle\right) = \left(\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle\right) \otimes \left(\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle\right)$$

Sin embargo, el estado:

$$|\psi\rangle = \frac{1}{\sqrt{2}}\left(|0,1\rangle + |1,0\rangle\right)$$

es un estado enredado porque el sistema de ecuaciones:

$$ac = 0 \quad : \quad ad = 1/\sqrt{2} \quad : \quad bc = 1/\sqrt{2} \quad : \quad bd = 0$$

no tiene solucion.

El análisis del comportamiento de un sistema cuántico en función de su medida permite observar si el sistema se encuentra o no en un estado enredado. Un sistema se encuentra en un estado enredado si la medida de uno de sus componentes afecta la medida de los otros, y el sistema se encuentra en un estado no enredado si esto no sucede. Así, para el estado no entrelazado:

$$|\psi\rangle = 1/2\ (|0, 0\rangle - |0, 1\rangle + |1, 0\rangle - |1, 1\rangle)$$

se obtienen las siguientes probalidades de medida sobre el primer qubit (P_1) y sobre el segundo qubit (P_2):

$$P_1(|0\rangle) = P_1(|1\rangle) = P_2(|0\rangle) = P_2(|1\rangle) = \frac{1}{2}$$

Por lo tanto, y según lo ya visto, si al medir el primer qubit obtenemos el estado $|0\rangle$, el sistema evoluciona hacia el nuevo estado normalizado:

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

Y si al medir el primer qubit obtenemos el estado $|1\rangle$, el sistema evoluciona hacia el nuevo estado normalizado:

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

En ambos casos las probabilidades de medida sobre el segundo qubit son:

$$P_2\ (|0\rangle) = P_2\ (|1\rangle) = 1/2.$$

de donde se infiere que la medida del primer qubit no afecta a la medida del segundo qubit.

Sin embargo, para el estado entrelazado:

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

las probabilidades de medida son:

$$P_1(|0\rangle)=P_1(|1\rangle)=P_2(|0\rangle)=P_2(|1\rangle)=\frac{1}{2}$$

Pero si al medir el primer qubit obtenemos $|0\rangle$ el sistema evoluciona hacia el nuevo estado normalizado:

$$|\psi\rangle = |0,1\rangle$$

y las probabilidades de medida del segundo qubit son: $P_2\ |0\rangle = 0$, $P_2\ |1\rangle = 1$. Sin embargo, si al medir el primer qubit obtenemos $|1\rangle$ el sistema evoluciona hacia el nuevo estado normalizado:

$$|\psi\rangle = |1,0\rangle$$

y las probabilidades de medida del segundo qubit son: $P_2\ |0\rangle = 1$, $P_2\ |1\rangle = 0$. En este caso la medida del primer qubit sí que afecta a la medida del segundo qubit.

--..--

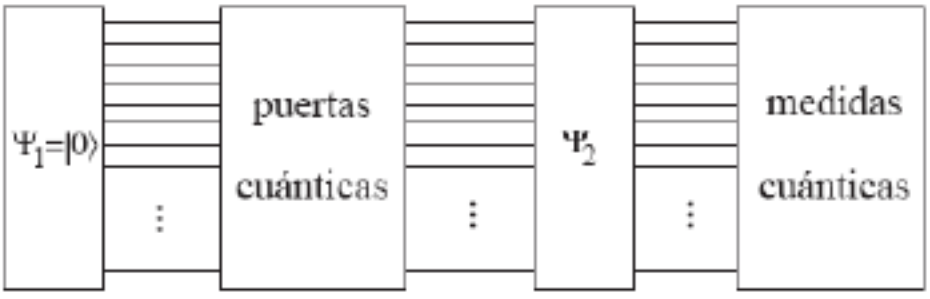
CONSTRUCCIÓN DE ALGORITMOS CON QUBITS

En el modelo cuántico de computación un algoritmo es un mecanismo para manipular n-qubits. Uno de los dos posibles mecanismos para hacerlo es medir qubits. El otro consiste en transformar un estado inicial ψ_1 en su correspondiente estado final ψ_2. La evolución o dinámica de un n-qubit es determinada por un operador unitario U sobre el espacio de Hilbert, este operador es denominado operador de evolución. Si llamamos U a la función de Vn $\rightarrow$ Vn tal que:

$$U\,\psi_1 = \psi_2$$

entonces la aplicación U transforma estados en estados, es decir, conserva la norma y, según los postulados de la mecánica cuántica, es lineal. Por tanto, U sólo puede ser una transformación unitaria. Un operador es unitario si su adjunto es igual a su inverso, y puede expresarse como[1]: $U^{\dagger}\,U = I$

Pero no podemos esperar que un ordenador cuántico sea capaz de aplicar una transformación unitaria genérica. Por lo tanto, deberemos describirla como una secuencia de transformaciones unitarias elementales que se denominan puertas cuánticas. En definitiva un algoritmo cuántico es una secuencia finita de puertas y medidas cuánticas.



Un algoritmo cuántico es una secuencia finita de puertas y medidas cuánticas.

[1] Usaremos indistintamente en lo que resta de texto las notaciones: $U^{\dagger}$, U', U^*

En la definición de algoritmo cuántico se han incluido dos restricciones. La primera afecta al estado inicial, que siempre será el mismo: $\psi_1 = |0\rangle$. La segunda consiste en que las puertas y las medidas cuánticas no se pueden alternar. En primer lugar se aplica una secuencia de puertas cuánticas, y a continuación una secuencia de medidas cuánticas. Estas restricciones simplifican los algoritmos cuánticos y no afectan al modelo de computación, pues todos los algoritmos cuánticos se pueden convertir en algoritmos cuánticos equivalentes que respetan estas restricciones.

Imposibilidad de Copiar Qubits

Una gran diferencia entre los modelos de computación cuántico y clásico está relacionada con la naturaleza de las puertas lógicas. Mientras que las puertas cuánticas son biyectivas por ser transformaciones unitarias, las puertas lógicas en general no lo son. La consecuencia más importante de esta propiedad de las puertas cuánticas es que los estados cuánticos no se pueden copiar. Para copiar un n-qubit bastaría encontrar una transformación unitaria U que cumpliese:

$$U\,(|x\rangle \otimes |0\rangle) = |x\rangle \otimes |x\rangle \quad \forall\ 0 \leq x < 2^n$$

donde los dos registros tienen n qubits. Esta transformación, si existiese, tendría que cumplir:

$$U\,(\psi \otimes |0\rangle) = \psi \otimes \psi$$

Pero la transformación U no existe para todo n-qubit ψ. Efectivamente, sea U una transformación unitaria en un espacio de dimensión 2^n tal que:

$$U\,(|a\rangle \otimes |0\rangle) = |a\rangle \otimes |a\rangle \qquad \text{y} \qquad U\,(|b\rangle \otimes |0\rangle) = |b\rangle \otimes |b\rangle$$

$$a \neq b\ ;\ \ 0 \leq a\ ,\ b < 2^n$$

Sea el n-qubit

$$\psi = (1/\sqrt{2})\ (|a\rangle + |b\rangle)$$

Entonces

$$U(\psi \otimes |0\rangle) = (1/\sqrt{2})\,(U(|a\rangle \otimes |0\rangle) + U(|b\rangle \otimes |0\rangle)) =$$

$$= (1/\sqrt{2})(\,|a\rangle \otimes |a\rangle + |b\rangle \otimes |b\rangle\,) \neq \psi \otimes \psi$$

Operaciones con Qubits

Sea $|\psi(t)\rangle = |\,x_1\,, \ldots, x_n\,\rangle$ un n-qubit. Podemos establecer que la evolución del sistema cuántico tras la aplicación del operador U en un paso de computación está dada por:

$$U\,|\psi(0)\rangle \rightarrow |\psi(1)\rangle$$

y en general, la evolución de "m" pasos de computación está dada por:

$$U^m\,|\psi(0)\rangle \rightarrow |\psi(m)\rangle$$

En el contexto de la computación cuántica un operador de evolución que opera sobre un n-qubit, corresponde a una matriz unitaria de dimensión 2^n. Por otra parte, hemos dicho ya que la computación es la creación de conjuntos de símbolos (resultados) a partir de ciertos conjuntos de símbolos iniciales (o datos). Si interpretamos los símbolos como objetos físicos, la computación correspondería a la evolución de los estados de los sistemas. Ya hemos visto que, en computación cuántica, estas evoluciones se materializan a partir de los operadores unitarios. Pero resulta, y esto es muy importante, que tales operadores unitarios no son más que representaciones matriciales de puertas lógicas reversibles, o compuertas cuánticas, con las que podemos construir circuitos cuánticos.

A diferencia de las compuertas lógicas convencionales, que pueden operar de n-bits en m-bits, las compuertas cuánticas deben operar de n-qubits en n-

qubits. Esta es una condición necesaria pero no suficiente para satisfacer la propiedad de reversibilidad de las compuertas cuánticas.

La reversibilidad de las puertas cuánticas es una consecuencia de la naturaleza unitaria de los operadores que las implementan. Sea Uf la matriz unitaria asociada a una puerta "f" entonces, para cualesquiera estados |x⟩ , |y⟩ se obtiene:

$$U_f|x\rangle = |y\rangle \Rightarrow U_f^{\dagger} U_f |x\rangle = U_f^{\dagger}|y\rangle \Rightarrow |x\rangle = U_f^{\dagger}|y\rangle$$

Lo que significa que a partir de la información de salida es posible obtener la información de entrada. Además, a partir de una función "f" de "n" bits en "m" bits se puede construir una función reversible, $f_{reversible}$, de "m + n" bits en "m + n" bits de acuerdo con el siguiente procedimiento:

$$f : x \to f(x) \Rightarrow f_{reversible} : (x, y) \to (x, y \oplus f(x))$$

De este modo, una función "f" puede ser implementada por un circuito cuántico U_f , cumpliendo las condiciones de reversibilidad exigidas a éste, si U_f realiza la transformación:

$$U\ |x, y\rangle \to |x, y \oplus f(x)\rangle$$

Cada compuerta cuántica de n-qubits puede ser representada por una matriz unitaria de dimensión 2^n , en donde la transformación realizada por la compuerta cuántica es realizada por el operador matriz asociado a ella.

Teniendo en cuenta la descripción de la transformación que realiza una compuerta cuántica sobre los elementos de la base del espacio, la matriz unitaria asociada a ella se obtiene a partir del siguiente procedimiento:

- Las filas de la matriz corresponden a los vectores base de entrada.
- Las columnas de la matriz corresponden a los vectores base de salida.

- La posición (j, i) de la matriz corresponde, cuando el i-ésimo vector base es la entrada a la compuerta, al coeficiente del j-ésimo vector base en la salida de la compuerta.

Transformaciones Unitarias de un Qubit

Las compuertas cuánticas que operan sobre un qubit (un qubit de entrada y un qubit de salida) tienen asociadas matrices 2 × 2. Por otra parte, ya hemos visto que para representar vectorialmente el ket 0 y el ket 1 el criterio es:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Veremos ahora cómo podemos operar con algunas de estas puertas. Empezaremos con la "identidad" (I) que, aunque su comportamiento no modifica el qubit sobre el que actúa, servirá para ilustrar el procedimiento de construcción de la matriz unitaria asociada.

Representación de la operación de identidad.

La transformación que efectúa esta puerta y su matriz unitaria se ilustran a continuación.

$$U_{\text{identidad}} |0\rangle \rightarrow |0\rangle \qquad : \qquad U_{\text{identidad}} |1\rangle \rightarrow |1\rangle$$

$$\begin{array}{c|cc} & |0\rangle & |1\rangle \\ \hline |0\rangle & 1 & 0 \\ |1\rangle & 0 & 1 \end{array} \qquad \rightarrow \qquad U_{\text{identidad}} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Por supuesto, el comportamiento de la matriz unitaria de la identidad es el siguiente:

$$U_{identidad}|0\rangle = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \times \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = |0\rangle : U_{identidad}|1\rangle = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \times \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = |1\rangle$$

Veamos ahora el comportamiento de la puerta de Hadamard (H), que transforma un qubit en una superposición de los elementos de la base {|0⟩,|1⟩}. La descripción y las transformaciones que realiza la puerta de Hadamard se ilustran a continuación.

$$U_{hadamard}|0\rangle \rightarrow \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right) : U_{hadamard}|1\rangle \rightarrow \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right)$$

$$\begin{array}{c|cc} & |0\rangle & |1\rangle \\ \hline |0\rangle & \frac{+1}{\sqrt{2}} & \frac{+1}{\sqrt{2}} \\ |1\rangle & \frac{+1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{array} \quad \rightarrow \quad U_{hadamard} = \frac{1}{\sqrt{2}} \begin{pmatrix} +1 & +1 \\ +1 & -1 \end{pmatrix}$$

$$U_{hadamard}|0\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} +1 & +1 \\ +1 & -1 \end{pmatrix} \times \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right)$$

$$U_{hadamard}|1\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} +1 & +1 \\ +1 & -1 \end{pmatrix} \times \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} +1 \\ -1 \end{pmatrix} = \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right)$$

Otras puertas interesantes son la "negación" (N), el "cambio de fase" (Z) y las "negación con cambio de fase" (Y), que describimos a continuación.

$$\text{Negación} : \text{N} \rightarrow N|0\rangle = |1\rangle \wedge N|1\rangle = |0\rangle \rightarrow N = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

$$\text{Cambio de fase} : \text{Z} \rightarrow Z|0\rangle = |0\rangle \wedge Z|1\rangle = -|1\rangle \rightarrow Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

$$\text{Negación con cambio de fase} : \text{Y} \rightarrow Y|0\rangle = -|1\rangle \wedge Y|1\rangle = |0\rangle \rightarrow Y = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

Las matrices de las transformaciones Identidad (I), Negación (N), y las matrices asociadas a las transformaciones $-iY$ y Z, se conocen con el nombre de matrices de Pauli -ya mencionadas anteriormente- y se utilizan para transportar estados cuánticos y para construir códigos correctores cuánticos. En el diseño de algoritmos cuánticos las transformaciones más importantes son H y N.

El ejemplo siguiente ilustra cómo obtener la transformación que realiza una puerta cuántica sobre los elementos de la base mediante el uso de puertas cuánticas parametrizadas. Sea:

$$U(\theta) = \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & \sin\left(\frac{\theta}{2}\right) \\ -\sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{pmatrix}$$

El comportamiento esta matriz sobre los elementos de la base {|0⟩ , |1⟩} es:

$$U(\theta)|0\rangle = \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & \sin\left(\frac{\theta}{2}\right) \\ -\sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{pmatrix} \times \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) \\ -\sin\left(\frac{\theta}{2}\right) \end{pmatrix} = \cos\left(\frac{\theta}{2}\right)|0\rangle - \sin\left(\frac{\theta}{2}\right)|1\rangle$$

$$U(\theta)|1\rangle = \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & \sin\left(\frac{\theta}{2}\right) \\ -\sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{pmatrix} \times \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} \sin\left(\frac{\theta}{2}\right) \\ \cos\left(\frac{\theta}{2}\right) \end{pmatrix} = \sin\left(\frac{\theta}{2}\right)|0\rangle + \cos\left(\frac{\theta}{2}\right)|1\rangle$$

La transformación de la puerta cuántica U (θ) es la siguiente:

$$U(\theta)|0\rangle \rightarrow \cos\left(\frac{\theta}{2}\right)|0\rangle - \sin\left(\frac{\theta}{2}\right)|1\rangle : U(\theta)|1\rangle \rightarrow \sin\left(\frac{\theta}{2}\right)|0\rangle + \cos\left(\frac{\theta}{2}\right)|1\rangle$$

Evidentemente, U(θ) puede actuar sobre estado qubitales diversos. Al respecto, es de gran utilidad la siguiente transformación:

$$U(\theta)\left(\cos\left(\frac{\theta}{2}\right)|0\rangle - \sin\left(\frac{\theta}{2}\right)|1\rangle\right) = \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & \sin\left(\frac{\theta}{2}\right) \\ -\sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{pmatrix} \times \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) \\ -\sin\left(\frac{\theta}{2}\right) \end{pmatrix} = \begin{pmatrix} \cos(\theta) \\ -\sin(\theta) \end{pmatrix} =$$

$$= \cos(\theta)|0\rangle - \sin(\theta)|1\rangle$$

Esta expresión se obtiene fácilmente sin más que recordar las relaciones trigonométricas:

- sin (a+b) = sin (a) cos (b) + cos (a) sin (b)
- sin (a-b) = sin (a) cos (b) - cos (a) sin (b)
- cos (a+b) = cos (a) cos (b) - sin (a) sin (b)
- cos (a-b) = cos (a) cos (b) + sin (a) sin (b)
- sin (2a) = 2 sin (a) cos (a)
- cos (2a) = $\cos^2$ (a) - $\sin^2$ (a)

Por otra parte, decíamos que las matrices de transformación, operadores, de la computación cuántica eran unitarias. En general, la naturaleza de la matriz asociada a una puerta cuántica M permite construir una nueva puerta cuántica $M^{\dagger}$ cuya matriz es la matriz hermítica conjugada $M^{\dagger}$ de M . En esta circunstancia reside la propiedad de reversibilidad de las compuertas cuánticas.

Para verlo, estudiaremos la matriz hermítica conjugada de la matriz U (θ) representada anteriormente y las transformaciones que realiza la correspondiente puerta U † (θ).

$$U^{\dagger}(\theta)|0\rangle \rightarrow \cos\left(\frac{\theta}{2}\right)|0\rangle + \sin\left(\frac{\theta}{2}\right)|1\rangle : U^{\dagger}(\theta)|1\rangle \rightarrow -\sin\left(\frac{\theta}{2}\right)|0\rangle + \cos\left(\frac{\theta}{2}\right)|1\rangle$$

$$U^{\dagger}(\theta) = \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & -\sin\left(\frac{\theta}{2}\right) \\ \sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{pmatrix}$$

Para la construcción de algunos circuitos cuánticos son muy útiles las siguientes transformaciones realizadas con la puerta U † (θ):

$$U^{\dagger}(\theta)\left(\cos(\theta)|0\rangle - \sin(\theta)|1\rangle\right) = \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & -\sin\left(\frac{\theta}{2}\right) \\ \sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{pmatrix} \times \begin{pmatrix} \cos(\theta) \\ -\sin(\theta) \end{pmatrix} = \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) \\ -\sin\left(\frac{\theta}{2}\right) \end{pmatrix}$$

$$U^{\dagger}(\theta)\left(\cos\left(\frac{\theta}{2}\right)|0\rangle - \sin\left(\frac{\theta}{2}\right)|1\rangle\right) = \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & -\sin\left(\frac{\theta}{2}\right) \\ \sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{pmatrix} \times \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) \\ -\sin\left(\frac{\theta}{2}\right) \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = |0\rangle$$

Al igual que antes, la obtención de estas expresiones requiere el empleo de las relaciones trigonométricas ya mencionadas.

Con lo visto hasta ahora ya estamos en condiciones de construir una matriz unitaria 2 × 2 generalizada. De hecho, toda matriz unitaria 2 × 2 puede ser factorizada con las cuatro matrices representadas a continuación:

$$U(\alpha,\beta,\delta,\theta)=\begin{pmatrix} e^{i\left(\delta+\frac{\alpha}{2}+\frac{\beta}{2}\right)}\cos\left(\frac{\theta}{2}\right) & e^{i\left(\delta+\frac{\alpha}{2}-\frac{\beta}{2}\right)}\sin\left(\frac{\theta}{2}\right) \\ -e^{i\left(\delta-\frac{\alpha}{2}+\frac{\beta}{2}\right)}\sin\left(\frac{\theta}{2}\right) & e^{i\left(\delta-\frac{\alpha}{2}-\frac{\beta}{2}\right)}\cos\left(\frac{\theta}{2}\right) \end{pmatrix}=$$

$$=\begin{pmatrix} e^{i\delta} & 0 \\ 0 & e^{i\delta} \end{pmatrix}\times\begin{pmatrix} e^{i\frac{\alpha}{2}} & 0 \\ 0 & e^{-i\frac{\alpha}{2}} \end{pmatrix}\times\begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & \sin\left(\frac{\theta}{2}\right) \\ -\sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{pmatrix}\times\begin{pmatrix} e^{i\frac{\beta}{2}} & 0 \\ 0 & e^{-i\frac{\beta}{2}} \end{pmatrix}$$

Terminaremos nuestra discusión sobre transformaciones unitarias de un qubit definiendo la matriz $R_y(\theta)$ que emplearemos en la construcción de algún circuito cuántico.

$$R_y(\theta)=\begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & \sin\left(\frac{\theta}{2}\right) \\ -\sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{pmatrix}$$

Transformaciones Unitarias de 2-qubits

Trabajaremos ahora un poco sobre sistemas de dos qubits, pero antes recordaremos que un 2-qubit $|x\ y\rangle$ se construye como $|x\rangle \otimes |y\rangle$. Por lo tanto, si consideramos los vectores de la base $|0\rangle$ y $|1\rangle$ resulta que:

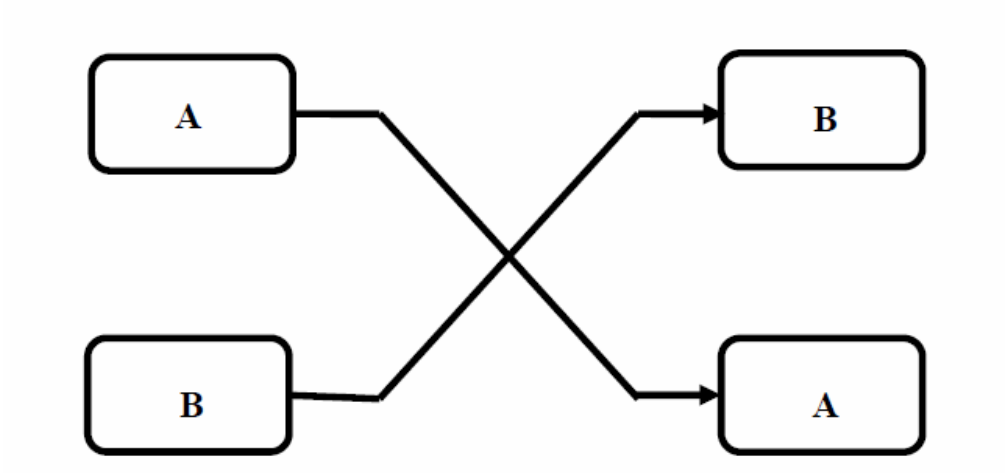
$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} : |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$|0\rangle \otimes |0\rangle = |00\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \Leftrightarrow |0\rangle \otimes |1\rangle = |01\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

$$|1\rangle \otimes |0\rangle = |10\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \Leftrightarrow |1\rangle \otimes |1\rangle = |11\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Por lo tanto, las matrices que realizan transformaciones unitarias en sistemas de dos qubits deben ser matrices de dimensión 4×4.

Volvamos un momento hacia atrás y recordemos algunas de las cuestiones tratadas cuando hablamos sobre reversibilidad. Una operación trivial que discutíamos era la operación de intercambio (EX), que se limitaba a intercambiar el estado de dos líneas según el esquema siguiente:



Operación de intercambio (EX).

La tabla de verdad asociada a esta puerta reversible es:

EX	*A*	*B*	*C*	*D*
	0	0	0	0
	0	1	1	0
	1	0	0	1
	1	1	1	1

En donde A y B representan las entradas mientras que C y D representan las salidas. Con bits convencionales el comportamiento de EX es el siguiente:

$$\text{EX}\,(0,0) \rightarrow (0,0) : \text{EX}\,(0,1) \rightarrow (1,0) : \text{EX}\,(1,0) \rightarrow (0,1) : \text{EX}\,(1,1) \rightarrow (1,1)$$

Sustituyamos ahora los bits por qubits y analicemos la transformación de la figura.

Representación de la operación de intercambio, EX.

$$U_{EX}|00\rangle \rightarrow |00\rangle : U_{EX}|01\rangle \rightarrow |10\rangle : U_{EX}|10\rangle \rightarrow |01\rangle : U_{EX}|11\rangle \rightarrow |11\rangle$$

En términos de matrices unitarias las expresiones anteriores no son otra cosa que lo siguiente:

EX	$\|00\rangle$	$\|01\rangle$	$\|10\rangle$	$\|11\rangle$
$\|00\rangle$	1	0	0	0
$\|01\rangle$	0	0	1	0
$\|10\rangle$	0	1	0	0
$\|11\rangle$	0	0	0	1

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

$$Claramente: U_{EX} \times U_{EX} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = U_I$$

$$Además: U_{EX}^{\dagger} = U_{EX}$$

Las dos últimas condiciones nos aseguran la reversibilidad de la puerta y la consistencia con los postulados de la mecánica cuántica.

Obviamente:

$$U_{EX}|00\rangle = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} = |00\rangle \Leftrightarrow U_{EX}|01\rangle = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} = |10\rangle \Leftrightarrow$$

$$\Leftrightarrow U_{EX}|10\rangle = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} = |01\rangle \Leftrightarrow U_{EX}|11\rangle = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} = |11\rangle$$

Otra puerta que también habíamos discutido al hablar sobre reversibilidad era la puerta CN, o puerta CONTROLLED-NOT, para la cual la tabla de verdad es la que se muestra a continuación:

| A ⟩	| B ⟩	| A' ⟩	| B' ⟩
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

Otra vez la tabla de verdad de CN.

Decíamos también que, claramente, se puede interpretar $|B_{out}\rangle$ como la salida de una puerta XOR con entradas $|A_{in}\rangle$ y $|B_{in}\rangle$: $|B_{out}\rangle$ = XOR (A_{in},B_{in})... sin embargo el dispositivo no es el mismo ya que la puerta CN genera dos salidas en lugar de una. Exploremos esto en términos cuánticos:

$$U_{xor}\,|x, y\rangle \rightarrow |x, x\oplus y\rangle$$

$$\begin{array}{c|cccc} U_{XOR} & |00\rangle & |01\rangle & |10\rangle & |11\rangle \\ \hline |00\rangle & 1 & 0 & 0 & 0 \\ |01\rangle & 0 & 1 & 0 & 0 \\ |10\rangle & 0 & 0 & 0 & 1 \\ |11\rangle & 0 & 0 & 1 & 0 \end{array} \rightarrow U_{XOR} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

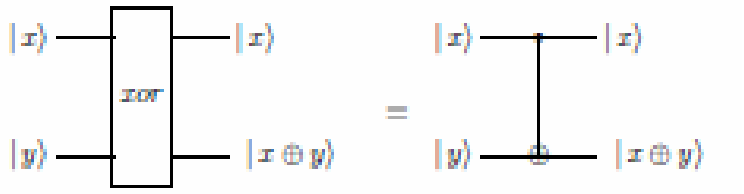
$$U_{xor}\,|0,0\rangle \rightarrow |0, 0 \oplus 0\rangle = |0,0\rangle \;:\; U_{xor}\,|0,1\rangle \rightarrow |0, 0 \oplus 1\rangle = |0,1\rangle$$

$$U_{xor}\,|1,0\rangle \rightarrow |1, 1 \oplus 0\rangle = |1,1\rangle \;:\; U_{xor}\,|1,1\rangle \rightarrow |1, 1 \oplus 1\rangle = |1,0\rangle$$

$$U_{XOR}|00\rangle = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \times \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} = |00\rangle : U_{XOR}|01\rangle = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \times \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} = |01\rangle$$

$$U_{XOR}|10\rangle = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \times \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} = |11\rangle : U_{XOR}|11\rangle = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \times \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} = |10\rangle$$

Esta puerta es representada normalmente por uno cualquiera de los circuitos de la figura siguiente.



Representación cuántica de la puerta CN.

La puerta CN también puede actuar sobre un qubit formado por cualquier combinación lineal de qubits, y su comportamiento es similar al descrito anteriormente, de forma que el cambio o no del segundo qubit es controlado por el primer qubit, como se aprecia en el ejemplo siguiente.

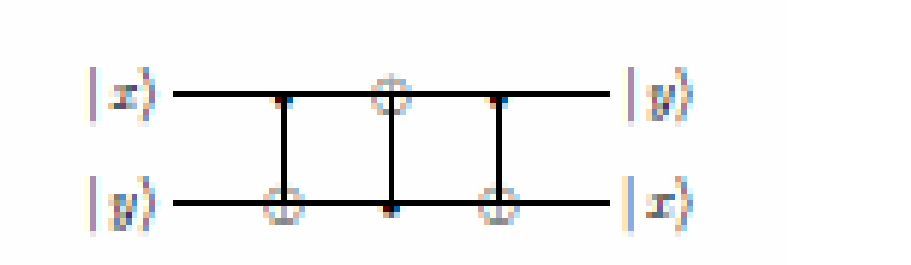
$$U_{XOR}\left(|0\rangle \otimes \left(a|0\rangle + b|1\rangle\right)\right) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \times \left(\begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} a \\ b \end{pmatrix}\right) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \times \begin{pmatrix} a \\ b \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} a \\ b \\ 0 \\ 0 \end{pmatrix} = \left(\begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} a \\ b \end{pmatrix}\right) = |0\rangle \otimes \left(a|0\rangle + b|1\rangle\right)$$

$$U_{XOR}\left(|1\rangle \otimes \left(a|0\rangle + b|1\rangle\right)\right) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \times \left(\begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} a \\ b \end{pmatrix}\right) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \times \begin{pmatrix} 0 \\ 0 \\ a \\ b \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ b \\ a \end{pmatrix} = \left(\begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} b \\ a \end{pmatrix}\right) = |1\rangle \otimes \left(b|0\rangle + a|1\rangle\right)$$

Como se observa en el ejemplo, si el primer qubit es cero el segundo qubit no cambia; pero si el primer qubit es uno el segundo qubit intercambia sus coeficientes.

Construcción de Circuitos Cuánticos

Acabamos de ver la transformación unitaria de intercambio de qubits (EX). A continuación construiremos un circuito cuántico que realiza esta operación. La implementación la haremos con tres puertas CN que, como ya sabemos equivale a la función XOR cuántica. El circuito correspondiente se ilustra en la figura.



Circuito cuántico de una puerta de intercambio construida a partir de tres puertas CN.

El funcionamiento de este circuito es el siguiente:

$$|x, y\rangle \rightarrow |x, x \oplus y\rangle \rightarrow |(x \oplus y) \oplus x, x \oplus y\rangle = |y, x \oplus y\rangle \rightarrow$$
$$\rightarrow |y, y \oplus (x \oplus y)\rangle = |y, x\rangle$$

Para construir la matriz unitaria 4 × 4 que representa el intercambio de dos qubits procedemos de la siguiente manera:

- El primer subcircuito realiza la transformación: $|x, y\rangle \rightarrow |x, x \oplus y\rangle$ que se corresponde con la puerta cuántica XOR cuya matriz unitaria es:

$$m_1 = U_{XOR} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

- El segundo subcircuito realiza la transformación $|x, y\rangle \rightarrow |y \oplus x, y\rangle$. Dado que quien controla ahora es el segundo qubit, la transformación que realiza el subcircuito y la matriz unitaria asociada ella están dadas por:

$$|0,0\rangle \rightarrow |0,0\rangle : |0,1\rangle \rightarrow |1,1\rangle : |1,0\rangle \rightarrow |1,0\rangle : |1,1\rangle \rightarrow |0,1\rangle$$

$$m_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

- El tercer subcircuito es similar al primer subcircuito por la tanto la matriz m_3 es igual a la matriz m_1 . La matriz U_{EX} correspondiente al circuito de intercambio de dos qubits será, entonces:

$$U_{EX} = (m_1 \times m_2) \times m_3 =$$

$$= \left[\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \times \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \right] \times \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} =$$

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

Este resultado es exactamente el mismo que habíamos obtenido analíticamente para la transformación U_{EX}.

Finalmente analizaremos el comportamiento cuántico de la puerta reversible CCN, o puerta de Toffoli, ya vista cuando discutíamos sobre computación reversible. Antes, sin embargo, recordaremos el carácter universal de la puerta de Toffoli.

Representación cuántica de CCN o puerta de Toffoli.

El conjunto de puertas lógicas {AND y NOT} , o la puerta NAND solita, es universal porque permite implementar cualquier función:

$$f : \{0,1\}^n \rightarrow \{0,1\}^m$$

Por otro lado, no es posible obtener un conjunto de puertas universales para funciones reversibles de la forma

$$f : \{0,1\}^n \rightarrow \{0,1\}^n$$

con compuertas reversibles de una línea ni con compuertas reversibles de 2 líneas. Sin embargo, sí que se puede utilizando la puerta de Toffoli que, como sabemos, opera sobre 3 líneas de la forma siguiente:

$$and : (x, y) \rightarrow x \wedge y \Rightarrow Toffoli : (x, y, z) \rightarrow (x, y, z \oplus (x \wedge y))$$

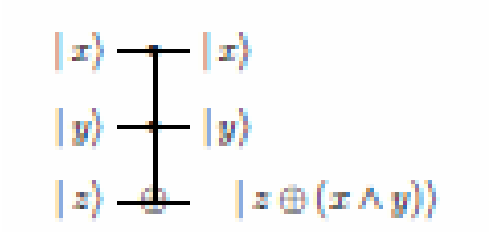
El comportamiento de la compuerta de Toffoli es descrito por la tabla que sigue, y su carácter de compuerta universal es resumido a continuación en donde se indica que dicha compuerta puede actuar como una compuerta AND o una compuerta NOT o una compuerta XOR o una compuerta Identidad.

x	y	z	$z \oplus (x \wedge y)$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

Tabla de la puerta de Toffoli.

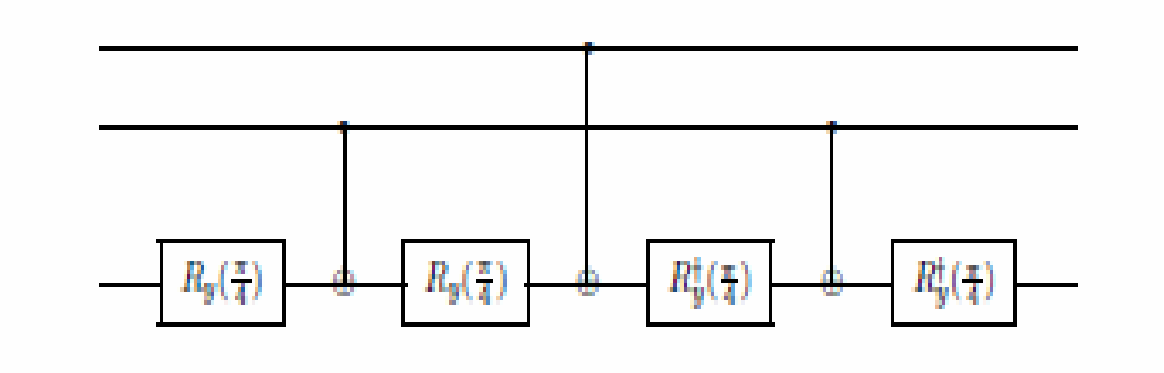
$$
\begin{aligned}
z \oplus (x \wedge y) &= x \wedge y \quad ; \quad z = 0 \\
&= x \oplus z \quad ; \quad y = 1 \\
&= \neg z \quad ; \quad x = y = 1 \\
&= z \quad ; \quad x = 0,\ y = 1
\end{aligned}
$$

La puerta de Toffoli puede ser vista como un circuito cuántico con su correspondiente matriz unitaria.



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

Aunque la puerta cuántica de Toffoli es una puerta de 3-qubits puede ser implementada por un circuito cuántico que sólo utiliza puertas cuánticas de 1-qubit y de 2-qubits. En particular, la compuerta cuántica de Toffoli puede ser descompuesta en un circuito cuántico formado por seis compuertas XOR y ocho compuertas de 1-qubit. Pero aún podemos simplificar más. Así, si podemos realizar un cambio de fase, la compuerta puede ser construida tal como se indica en la figura.



Circuito cuántico de una puerta de Toffoli con cambio de fase.

La implementación de la compuerta de Toffoli con compuertas cuánticas de 1-qubits y 2-qubits utiliza puertas Ry ($\pi/4$) y sus transpuestas conjugadas, cuyas matrices unitarias fueron descritas anteriormente, y compuertas XOR cuánticas. Analizaremos su comportamiento en tres de los 8 casos posibles.

1er caso: Entrada = $|0, 0, 0\rangle$

- Paso 0: Los valores iniciales de los kets $|x\rangle$, $|y\rangle$, $|z\rangle$ son $|0\rangle$, $|0\rangle$, $|0\rangle$ respectivamente, con lo cual se forma el estado cuántico inicial:

$$|0\rangle \otimes |0\rangle \otimes |0\rangle = |0,0,0\rangle$$

- Paso 1: La compuerta Ry_1 actúa sobre el ket $|z\rangle = |0\rangle$, según ya vimos anteriormente:

$$R_y\left(\frac{\pi}{4}\right)|0\rangle = \begin{pmatrix} \cos\left(\frac{\pi}{8}\right) & \sin\left(\frac{\pi}{8}\right) \\ -\sin\left(\frac{\pi}{8}\right) & \cos\left(\frac{\pi}{8}\right) \end{pmatrix} \times \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} \cos\left(\frac{\pi}{8}\right) \\ -\sin\left(\frac{\pi}{8}\right) \end{pmatrix} = \cos\left(\frac{\pi}{8}\right)|0\rangle - \sin\left(\frac{\pi}{8}\right)|1\rangle$$

El nuevo estado cuántico es :

$$|0\rangle \otimes |0\rangle \otimes \left\{\cos\left(\frac{\theta}{8}\right)|0\rangle - \sin\left(\frac{\theta}{8}\right)|1\rangle\right\} = \cos\left(\frac{\theta}{8}\right)|0,0,0\rangle - \sin\left(\frac{\theta}{8}\right)|0,0,1\rangle$$

- Paso 2: La compuerta XOR_1 efectua un XOR entre los ket de las líneas:

$|y\rangle = |0\rangle$, $|z\rangle = \cos(\pi/8)\,|0\rangle - \sin(\pi/8)|1\rangle$

$$|y\rangle \oplus |z\rangle = |0\rangle \oplus \left\{\cos\left(\frac{\pi}{8}\right)|0\rangle - \sin\left(\frac{\pi}{8}\right)|1\rangle\right\} = \cos\left(\frac{\pi}{8}\right)|0\rangle - \sin\left(\frac{\pi}{8}\right)|1\rangle$$

El nuevo estado cuántico es:

$$|0\rangle \otimes |0\rangle \otimes \left\{|0\rangle \oplus \left(\cos\left(\frac{\pi}{8}\right)|0\rangle - \sin\left(\frac{\pi}{8}\right)|1\rangle\right)\right\} = \cos\left(\frac{\pi}{8}\right)|0,0,0\rangle - \sin\left(\frac{\pi}{8}\right)|0,0,1\rangle$$

- Paso 3: La compuerta Ry_2 actúa sobre $|z\rangle = \cos(\pi/8)|0\rangle - \text{sen}(\pi/8)|1\rangle$ de la forma siguiente:

$$R_y\left(\frac{\pi}{4}\right)\left\{\cos\left(\frac{\pi}{8}\right)|0\rangle-\sin\left(\frac{\pi}{8}\right)|1\rangle\right\}=\begin{pmatrix}\cos\left(\frac{\pi}{8}\right) & \sin\left(\frac{\pi}{8}\right)\\ -\sin\left(\frac{\pi}{8}\right) & \cos\left(\frac{\pi}{8}\right)\end{pmatrix}\times\begin{pmatrix}\cos\left(\frac{\pi}{8}\right)\\ -\sin\left(\frac{\pi}{8}\right)\end{pmatrix}=$$

$$=\begin{pmatrix}\cos^2\left(\frac{\pi}{8}\right)-\sin^2\left(\frac{\pi}{8}\right)\\ -2\sin\left(\frac{\pi}{8}\right)\cos\left(\frac{\pi}{8}\right)\end{pmatrix}=\begin{pmatrix}\cos\left(\frac{\pi}{4}\right)\\ -\sin\left(\frac{\pi}{4}\right)\end{pmatrix}=\cos\left(\frac{\pi}{4}\right)\|0\rangle-\sin\left(\frac{\pi}{4}\right)\|1\rangle$$

El nuevo estado cuántico es:

$$|0\rangle\otimes|0\rangle\otimes\left\{\cos\left(\frac{\pi}{4}\right)\|0\rangle-\sin\left(\frac{\pi}{4}\right)\|1\rangle\right\}=\cos\left(\frac{\pi}{4}\right)\|0,0,0\rangle-\sin\left(\frac{\pi}{4}\right)\|0,0,1\rangle$$

- Paso 4: La compuerta XOR_2 efectúa un XOR entre |x⟩ y |z⟩:

$$|x\rangle\oplus|z\rangle=|0\rangle\oplus\left\{\cos\left(\frac{\pi}{4}\right)\|0\rangle-\sin\left(\frac{\pi}{4}\right)\|1\rangle\right\}=\cos\left(\frac{\pi}{4}\right)|0\rangle-\sin\left(\frac{\pi}{4}\right)\|1\rangle$$

El nuevo estado cuántico es:

$$|0\rangle\otimes|0\rangle\otimes\left\{\cos\left(\frac{\pi}{4}\right)\|0\rangle-\sin\left(\frac{\pi}{4}\right)\|1\rangle\right\}=\cos\left(\frac{\pi}{4}\right)\|0,0,0\rangle-\sin\left(\frac{\pi}{4}\right)\|0,0,1\rangle$$

- Paso 5: La compuerta Ry_3 (que es transpuesta conjugada) actúa sobre |z⟩ del siguiente modo:

$$R^+{}_y\left(\frac{\pi}{4}\right)\left\{\cos\left(\frac{\pi}{4}\right)|0\rangle-\sin\left(\frac{\pi}{4}\right)|1\rangle\right\}=\begin{pmatrix}\cos\left(\frac{\pi}{8}\right) & -\sin\left(\frac{\pi}{8}\right)\\ \sin\left(\frac{\pi}{8}\right) & \cos\left(\frac{\pi}{8}\right)\end{pmatrix}\times\begin{pmatrix}\cos\left(\frac{\pi}{4}\right)\\ -\sin\left(\frac{\pi}{4}\right)\end{pmatrix}=$$

$$=\begin{pmatrix}\cos\left(\frac{\pi}{8}\right)\cos\left(\frac{\pi}{4}\right)+\sin\left(\frac{\pi}{8}\right)\sin\left(\frac{\pi}{4}\right)\\ \sin\left(\frac{\pi}{8}\right)\cos\left(\frac{\pi}{4}\right)-\cos\left(\frac{\pi}{8}\right)\sin\left(\frac{\pi}{4}\right)\end{pmatrix}=\begin{pmatrix}\cos\left(\frac{\pi}{8}\right)\\ -\sin\left(\frac{\pi}{8}\right)\end{pmatrix}=\cos\left(\frac{\pi}{8}\right)\|0\rangle-\sin\left(\frac{\pi}{8}\right)\|1\rangle$$

El nuevo estado cuántico es:

$$|0\rangle \otimes |0\rangle \otimes \left\{ \cos\left(\frac{\pi}{8}\right)|0\rangle - \sin\left(\frac{\pi}{8}\right)|1\rangle \right\} = \cos\left(\frac{\pi}{8}\right)|0,0,0\rangle - \sin\left(\frac{\pi}{8}\right)|0,0,1\rangle$$

- Paso 6: La compuerta XOR_3 efectúa un XOR entre $|y\rangle$ y $|z\rangle$ del siguiente modo:

$$|y\rangle \oplus |z\rangle = |0\rangle \oplus \left\{ \cos\left(\frac{\pi}{8}\right)|0\rangle - \sin\left(\frac{\pi}{8}\right)|1\rangle \right\} = \cos\left(\frac{\pi}{8}\right)|0\rangle - \sin\left(\frac{\pi}{8}\right)|1\rangle$$

El nuevo estado cuántico es:

$$|0\rangle \otimes |0\rangle \otimes \left\{ \cos\left(\frac{\pi}{8}\right)|0\rangle - \sin\left(\frac{\pi}{8}\right)|1\rangle \right\} = \cos\left(\frac{\pi}{8}\right)|0,0,0\rangle - \sin\left(\frac{\pi}{8}\right)|0,0,1\rangle$$

- Paso 7: Finalmente, la compuerta Ry_4 (que es transpuesta conjugada) actúa sobre $|z\rangle$ del siguiente modo:

$$R^{+}{}_{y}\left(\frac{\pi}{4}\right)\left\{ \cos\left(\frac{\pi}{8}\right)|0\rangle - \sin\left(\frac{\pi}{8}\right)|1\rangle \right\} = \begin{pmatrix} \cos\left(\frac{\pi}{8}\right) & -\sin\left(\frac{\pi}{8}\right) \\ \sin\left(\frac{\pi}{8}\right) & \cos\left(\frac{\pi}{8}\right) \end{pmatrix} \times \begin{pmatrix} \cos\left(\frac{\pi}{8}\right) \\ -\sin\left(\frac{\pi}{8}\right) \end{pmatrix} =$$

$$= \begin{pmatrix} \cos^2\left(\frac{\pi}{8}\right) + \sin^2\left(\frac{\pi}{8}\right) \\ \sin\left(\frac{\pi}{8}\right)\cos\left(\frac{\pi}{8}\right) - \cos\left(\frac{\pi}{8}\right)\sin\left(\frac{v}{8}\right) \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = |0\rangle$$

El nuevo estado cuántico es:

$$|0\rangle \otimes |0\rangle \otimes |0\rangle = |0,0,0\rangle$$

Por lo tanto, hemos podido comprobar que: $U_{Toffoli}\ |000\rangle \rightarrow |000\rangle$

2º caso: Entrada = $|1, 0, 0\rangle$

Escribiremos la estructura de la puerta como una secuencia de transformaciones unitarias, de acuerdo con lo ilustrado en la figura anterior:

$U_{Toffoli} = Ry(\theta/4)_{|z\rangle} \times (|y\rangle \oplus |z\rangle) \times Ry(\theta/4)_{|z\rangle} \times (|x\rangle \oplus |z\rangle) \times R^{\dagger}y(\theta/4)_{|z\rangle} \times (|y\rangle \oplus |z\rangle) \times R^{\dagger}y(\theta/4)_{|z\rangle}$

Según esta estructura, y con la entrada $|1, 0,0\rangle$, los resultados intermedios son:

- Paso 0: $|1, 0,0\rangle$
- Paso 1: $\cos(\pi/8)\,|1, 0, 0\rangle - \sin(\pi/8)\,|1, 0, 1\rangle$
- Paso 2: $\cos(\pi/8)\,|1, 0, 0\rangle - \sin(\pi/8)\,|1, 0, 1\rangle$
- Paso 3: $\cos(\pi/4)\,|1, 0, 0\rangle - \sin(\pi/4)\,|1, 0, 1\rangle$
- Paso 4: $-\sin(\pi/4)\,|1, 0, 0\rangle + \cos(\pi/4)\,|1, 0, 1\rangle$
- Paso 5: $-\sin(3\pi/8)\,|1, 0, 0\rangle + \cos(3\pi/8)\,|1, 0, 1\rangle$
- Paso 6: $-\sin(3\pi/8)\,|1, 0, 0\rangle + \cos(3\pi/8)\,|1, 0, 1\rangle$
- Paso 7: $-|1, 0, 0\rangle$

Por lo tanto, hemos podido comprobar que: $U_{Toffoli}\ |100\rangle \rightarrow -|100\rangle$, lo que implica un cambio de fase.

3er caso: Entrada = $|1, 1, 1\rangle$

Este caso ilustra cómo la compuerta de Toffoli cambia el estado del tercer qubit.

- Paso 0: $|1, 1,1\rangle$
- Paso 1: $\sin(\pi/8)\,|1, 1, 0\rangle + \cos(\pi/8)\,|1, 1, 1\rangle$

- Paso 2: $\cos(\pi/8)\,|1, 1, 0\rangle + \sin(\pi/8)\,|1, 1, 1\rangle$
- Paso 3: $|1, 1, 0\rangle$
- Paso 4: $|1, 1, 1\rangle$
- Paso 5: $-\sin(\pi/8)\,|1, 1, 0\rangle + \cos(\pi/8)\,|1, 1, 1\rangle$
- Paso 6: $\cos(\pi/8)\,|1, 1, 0\rangle - \sin(\pi/8)\,|1, 1, 1\rangle$
- Paso 7: $|1, 1, 0\rangle$

Por lo tanto, hemos podido comprobar que: $U_{Toffoli}\,|1, 1, 1\rangle \rightarrow |1, 1, 0\rangle$

--..--

LA COMPUTADORA CUÁNTICA DE FEYNMAN

Ya estamos casi en condiciones de diseñar algunos algoritmos cuánticos o, en su defecto, de estudiar alguno de los algoritmos cuánticos más relevantes. Pero nos falta algo todavía. Disponemos de un modelo de computación cuántica, pero no tenemos aún ningún modelo de computadora sobre la cual ensayar nuestros algoritmos cuánticos. Trataremos de llenar este hueco estudiando el modelo de computadora cuántica propuesto por el creativo Feynman a principios de los años 80 del siglo XX.

Jugando con Matrices

Sea un sistema cuántico ideal, por ejemplo con átomos. El sistema puede estar en uno cualquiera de dos estados:

Arriba ($\uparrow$)	:	Estado excitado
Abajo ($\downarrow$)	:	Estado no excitado
Espín ($\uparrow$)	$\equiv$	Bit (1)
Espín ($\downarrow$)	$\equiv$	Bit (0)

Construimos nuestro dispositivo de computación a partir de estos átomos, uniéndolos unos a otros de una forma concreta. Sea una parte, o todo, el sistema un conjunto de átomos cada uno de los cuales está en uno cualquiera de los dos estados posibles. Esto representa un número que es la entrada. Dejamos que el sistema evolucione durante un tiempo $\langle t \rangle$. La evolución se efectúa de acuerdo con las leyes de la mecánica cuántica:

1. El sistema interacciona consigo mismo
2. Los átomos cambian de estado
3. Los $\langle 1 \rangle$ y los $\langle 0 \rangle$ se cambian

En un momento determinado tenemos un conjunto de átomos en ciertos estados que representan la salida del sistema. Sobre este planteamiento, Feynman propone su modelo de computadora, para el que utiliza las llamada matrices de aniquilación y de creación. Veamos qué cosa extraña es eso.

Sea una línea de computación "A" sobre la cual hay una puerta lógica que realiza alguna transformación unitaria, por ejemplo la Identidad o la Negación. Podemos definir las siguientes matrices:

$$\mathbf{a_1} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} : \mathbf{a_2} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} : \mathbf{a_3} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} : \mathbf{a_4} = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}$$

Sólo por jugar un poco, vamos a multiplicar de todas las maneras posibles las matrices anteriores:

$$\mathbf{a}_1 \times \mathbf{a}_2 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \times \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} = \mathbf{a}_2 \Leftrightarrow \mathbf{a}_2 \times \mathbf{a}_1 = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \times \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} = 0$$

$$\mathbf{a}_1 \times \mathbf{a}_3 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \times \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} = 0 \Leftrightarrow \mathbf{a}_3 \times \mathbf{a}_1 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \times \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} = 0$$

$$\mathbf{a}_1 \times \mathbf{a}_4 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \times \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} = 0 \Leftrightarrow \mathbf{a}_4 \times \mathbf{a}_1 = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \times \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} = \mathbf{a}_4$$

$$\mathbf{a}_2 \times \mathbf{a}_3 = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \times \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} = \mathbf{a}_2 \Leftrightarrow \mathbf{a}_3 \times \mathbf{a}_2 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \times \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} = 0$$

$$\mathbf{a}_2 \times \mathbf{a}_4 = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \times \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} = \mathbf{a}_1 \Leftrightarrow \mathbf{a}_4 \times \mathbf{a}_2 = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \times \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} = \mathbf{a}_3$$

$$\mathbf{a}_3 \times \mathbf{a}_4 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \times \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} = \mathbf{a}_4 \Leftrightarrow \mathbf{a}_4 \times \mathbf{a}_3 = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \times \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} = 0$$

Sigamos haciendo cosas con estas matrices, por ejemplo construir las matrices unitarias de la identidad, **I**, y de la negación, **N**:

$$\mathbf{I} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \mathbf{a_1} + \mathbf{a_3} = (\mathbf{a_2 a_4} + \mathbf{a_4 a_2}) = \mathbf{1}$$

$$\mathbf{N} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \mathbf{a_2} + \mathbf{a_4}$$

Calculemos esta vez las correspondientes matrices transpuestas conjugadas:

$$\mathbf{a_1} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \rightarrow \mathbf{a_1^*} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} = \mathbf{a_1} \Leftrightarrow \mathbf{a}_2 = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \rightarrow \mathbf{a}_2^* = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} = \mathbf{a}_4$$

$$\mathbf{a}_3 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \rightarrow \mathbf{a}_3^* = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} = \mathbf{a}_3 \Leftrightarrow \mathbf{a}_4 = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \rightarrow \mathbf{a}_4^* = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} = \mathbf{a}_2$$

Como vemos, $\mathbf{a_1}$ y $\mathbf{a_3}$ no cambian, mientras que $\mathbf{a_2}$ y $\mathbf{a_4}$ son matrices transpuestas conjugadas la una de la otra.

Veamos ahora qué pasa cuando aplicamos $\mathbf{a_2}$ y $\mathbf{a_4}$ a los estados $|0\rangle$ y $|1\rangle$:

$$\mathbf{a}_2|0\rangle = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \times \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} = 0 \Leftrightarrow \mathbf{a}_2|1\rangle = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \times \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = |0\rangle$$

$$\mathbf{a}_4|0\rangle = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \times \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = |1\rangle \Leftrightarrow \mathbf{a}_4|1\rangle = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \times \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} = 0$$

Las matrices $\mathbf{a_2}$ y $\mathbf{a_4}$ reciben el nombre de matrices de aniquilación y de creación respectivamente. Cuando $\mathbf{a_2}$ opera sobre un ket 0 no hace nada, mientras que si lo hace sobre un ket 1 lo convierte en ket 0. Análogamente, cuando $\mathbf{a_4}$ opera sobre un ket 1 no hace nada, mientras que si lo hace sobre un ket 0 lo convierte en ket 1. Por lo tanto la matriz de aniquilación asegura un estado $|0\rangle$ mientras que la matriz de creación asegura un estado $|1\rangle$.

¿Y qué podemos hacer con estas matrices de aniquilación y de creación? Antes de responder a esta cuestión, y dado que:

- Nos interesan las matrices de aniquilación y de creación
- Sabemos ya que la una es la transpuesta conjugada de la otra

llamaremos **a** a la matriz de aniquilación, y **a*** a la matriz de creación, de forma que:

$$aa^* |0\rangle = |0\rangle : aa^* |1\rangle = 0 : a^*a |0\rangle = 0 : a^*a |1\rangle = |1\rangle$$

Por consiguiente (aa*) indica que $|A\rangle = |0\rangle$ y (a*a) indica que $|A\rangle = |1\rangle$. Ahora podemos utilizar este criterio para representar en términos de matrices de aniquilación-creación el comportamiento de las puertas cuánticas. Por ejemplo, si recordamos que CN consta de dos líneas (A y B), que la línea A es la de control (y por lo tanto $|A_{in}\rangle = |A_{out}\rangle$), que si $|A\rangle = |0\rangle \rightarrow |B_{out}\rangle = |B_{in}\rangle$, y que si $|A\rangle = |1\rangle \rightarrow |B_{out}\rangle$ = NOT $|B_{in}\rangle$, entonces:

$$U_{CN} = aa^* + (a^*a)\,(b + b^*) = 1 - a^*a + (a^*a)\,(b + b^*) = 1 + a^*a\,(b + b^* - 1)$$

La expresión anterior puede interpretarse del siguiente modo: Si la línea A está a cero (aa*), el sistema no ejecuta ninguna acción de control sobre la línea B. Por el contrario, si la línea A está a uno (a*a), entonces en sistema invierte el estado de la línea B (b + b*). Interpretando ahora los signos "+" y "×" (que equivales a las conjunciones "o" e "y"): ' La línea A está a cero (aa*), o (+) la línea A está a uno (a*a) y entonces (×) negamos la línea B (b + b*) '

Demostraremos ahora la equivalencia de las expresiones anteriores.

$$\mathbf{aa^*} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \Leftrightarrow \mathbf{a^*a} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \Leftrightarrow \mathbf{1} - \mathbf{aa^*} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} - \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} = \mathbf{a^*a}$$

Por lo tanto, la expresión algebraica es correcta. Pero además:

$$\mathbf{aa}^{*}+(\mathbf{a}^{*}\mathbf{a})(\mathbf{b}+\mathbf{b}^{*})=\begin{pmatrix}1&0\\0&0\end{pmatrix}+\begin{pmatrix}0&0\\0&1\end{pmatrix}\times\begin{pmatrix}0&1\\1&0\end{pmatrix}=\begin{pmatrix}1&0\\0&0\end{pmatrix}+\begin{pmatrix}0&0\\1&0\end{pmatrix}=\begin{pmatrix}1&0\\1&0\end{pmatrix}$$

$$\mathbf{1}+(\mathbf{a}^{*}\mathbf{a})(\mathbf{b}+\mathbf{b}^{*}-\mathbf{1})=\begin{pmatrix}1&0\\0&1\end{pmatrix}+\begin{pmatrix}0&0\\0&1\end{pmatrix}\times\left[\begin{pmatrix}0&1\\1&0\end{pmatrix}-\begin{pmatrix}1&0\\0&1\end{pmatrix}\right]=\begin{pmatrix}1&0\\0&1\end{pmatrix}+\begin{pmatrix}0&0\\0&1\end{pmatrix}\times\begin{pmatrix}-1&1\\1&-1\end{pmatrix}=$$

$$=\begin{pmatrix}1&0\\0&1\end{pmatrix}+\begin{pmatrix}0&0\\1&-1\end{pmatrix}=\begin{pmatrix}1&0\\1&0\end{pmatrix}$$

Sigamos ahora con la puerta CCN, para la cual -recordamos- aplican las siguientes restricciones:

- $|A'\rangle = |A\rangle$
- $|B'\rangle = |B\rangle$
- Si $|A\rangle = |0\rangle \wedge |B\rangle = |0\rangle \rightarrow |C'\rangle = |C\rangle$
- Si $|A\rangle = |0\rangle \wedge |B\rangle = |1\rangle \rightarrow |C'\rangle = |C\rangle$
- Si $|A\rangle = |1\rangle \wedge |B\rangle = |0\rangle \rightarrow |C'\rangle = |C\rangle$
- Si $|A\rangle = |1\rangle \wedge |B\rangle = |1\rangle \rightarrow |C'\rangle = \neg |C\rangle$

Por lo tanto, su representación en términos de matrices de aniquilación y creación es la siguiente:

$$\mathbf{U}_{\mathbf{CCN}}=(\mathbf{aa}^{*})(\mathbf{bb}^{*})+(\mathbf{aa}^{*})(\mathbf{b}^{*}\mathbf{b})+(\mathbf{a}^{*}\mathbf{a})(\mathbf{bb}^{*})+(\mathbf{a}^{*}\mathbf{a})(\mathbf{b}^{*}\mathbf{b})(\mathbf{c}+\mathbf{c}^{*})=$$

$$=\mathbf{1}+(\mathbf{a}^{*}\mathbf{a})(\mathbf{b}^{*}\mathbf{b})(\mathbf{c}+\mathbf{c}^{*}-\mathbf{1})$$

Vamos ahora a encontrar la expresión matricial de la puerta de bifurcación, FO, en términos de operaciones de aniquilación-creación. Para ello tenemos que representar esta puerta como un circuito CN de dos líneas, A y B, en el que la línea A es la de control, y la entrada de B es siempre cero. Así las cosas:

- $|A\rangle = |A\rangle$
- $|B\rangle = |0\rangle$
- $|A'\rangle = |A\rangle$
- Si $|A\rangle = |0\rangle \rightarrow |B'\rangle = |B\rangle = |0\rangle$
- Si $|A\rangle = |1\rangle \rightarrow |B'\rangle = \text{NOT } |B\rangle = |1\rangle$

Recordamos que en la línea A: $\mathbf{aa^*} \rightarrow |0\rangle$ y que $\mathbf{a^*a} \rightarrow |1\rangle$ y que ocurre lo mismo en la línea B. Por lo tanto:

$$U_{F0} = (\mathbf{aa^*})(\mathbf{bb^*}) + (\mathbf{a^*a})(\mathbf{bb^*})(\mathbf{b} + \mathbf{b^*}) = (1 - \mathbf{a^*a})(\mathbf{bb^*}) + (\mathbf{a^*a})(\mathbf{bb^*})(\mathbf{b} + \mathbf{b^*}) =$$

$$= (\mathbf{bb^*}) - (\mathbf{a^*a})(\mathbf{bb^*}) + (\mathbf{a^*a})(\mathbf{bb^*})(\mathbf{b} + \mathbf{b^*}) = (\mathbf{bb^*}) + (\mathbf{a^*a})(\mathbf{bb^*})(\mathbf{b} + \mathbf{b^*} - \mathbf{1})$$

Comprobamos:

$$(\mathbf{b} + \mathbf{b}^*) = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \rightarrow (\mathbf{bb}^*) \times (\mathbf{b} + \mathbf{b}^*) = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \times \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \rightarrow$$

$$\rightarrow (\mathbf{a}^* \mathbf{a}) \times (\mathbf{bb}^*) \times (\mathbf{b} + \mathbf{b}^*) = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \times \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$$

Además :

$$(\mathbf{aa}^*) \times (\mathbf{bb}^*) = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \times \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \rightarrow$$

$$\rightarrow U_{FO_1} = (\mathbf{aa}^*) \times (\mathbf{bb}^*) + (\mathbf{a}^* \mathbf{a}) \times (\mathbf{bb}^*) \times (\mathbf{b} + \mathbf{b}^*) = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$$

Análogamente:

$$(\mathbf{b}+\mathbf{b}^*-\mathbf{1})=\begin{pmatrix}0&1\\1&0\end{pmatrix}-\begin{pmatrix}1&0\\0&1\end{pmatrix}=\begin{pmatrix}-1&1\\1&-1\end{pmatrix}\rightarrow$$

$$\rightarrow(\mathbf{bb}^*)\times(\mathbf{b}+\mathbf{b}^*-1)=\begin{pmatrix}1&0\\0&0\end{pmatrix}\times\begin{pmatrix}-1&1\\1&-1\end{pmatrix}=\begin{pmatrix}-1&1\\0&0\end{pmatrix}\rightarrow$$

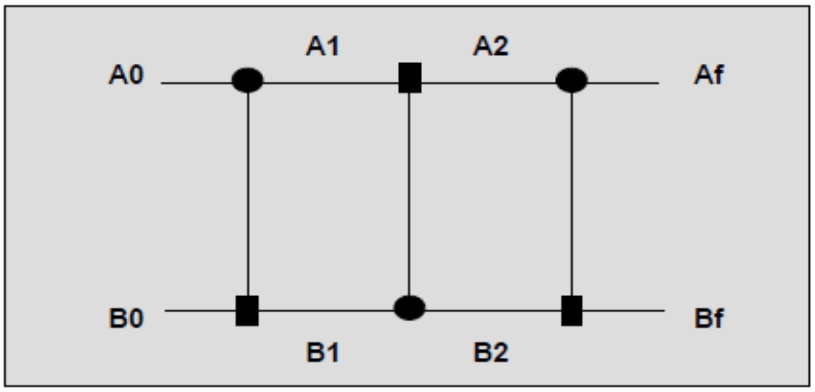
$$\rightarrow(\mathbf{a}^*\mathbf{a})\times(\mathbf{bb}^*)\times(\mathbf{b}+\mathbf{b}^*-1)=\begin{pmatrix}0&0\\0&1\end{pmatrix}\times\begin{pmatrix}-1&1\\0&0\end{pmatrix}=\begin{pmatrix}0&0\\0&0\end{pmatrix}\rightarrow$$

$$\rightarrow U_{FO_2}=(\mathbf{bb}^*)+(\mathbf{a}^*\mathbf{a})\times(\mathbf{bb}^*)\times(\mathbf{b}+\mathbf{b}^*-1)=\begin{pmatrix}1&0\\0&0\end{pmatrix}+\begin{pmatrix}0&0\\0&0\end{pmatrix}=\begin{pmatrix}1&0\\0&0\end{pmatrix}$$

Evidentemente ambas expresiones son equivalentes.

Nos quedan todavía por tratar dos puertas reversibles: el intercambio ⟨EX⟩, y el intercambio controlado o puerta de Fredkin. Sin embargo no desarrollaremos aquí sus expresiones matriciales. En este texto nos limitaremos a indicar cómo pueden obtenerse, sugiriendo –eso sí– las combinaciones de otras puertas reversibles que pueden llevarnos hasta la solución. Dejamos el resto como ejercicio para el lector estudioso.

El intercambio puede obtenerse mediante una arquitectura como la mostrada en la figura, y que combina tres puertas CN. La evolución de los valores que circulan por cada línea, y en cada paso, aparecen representados en la tabla.



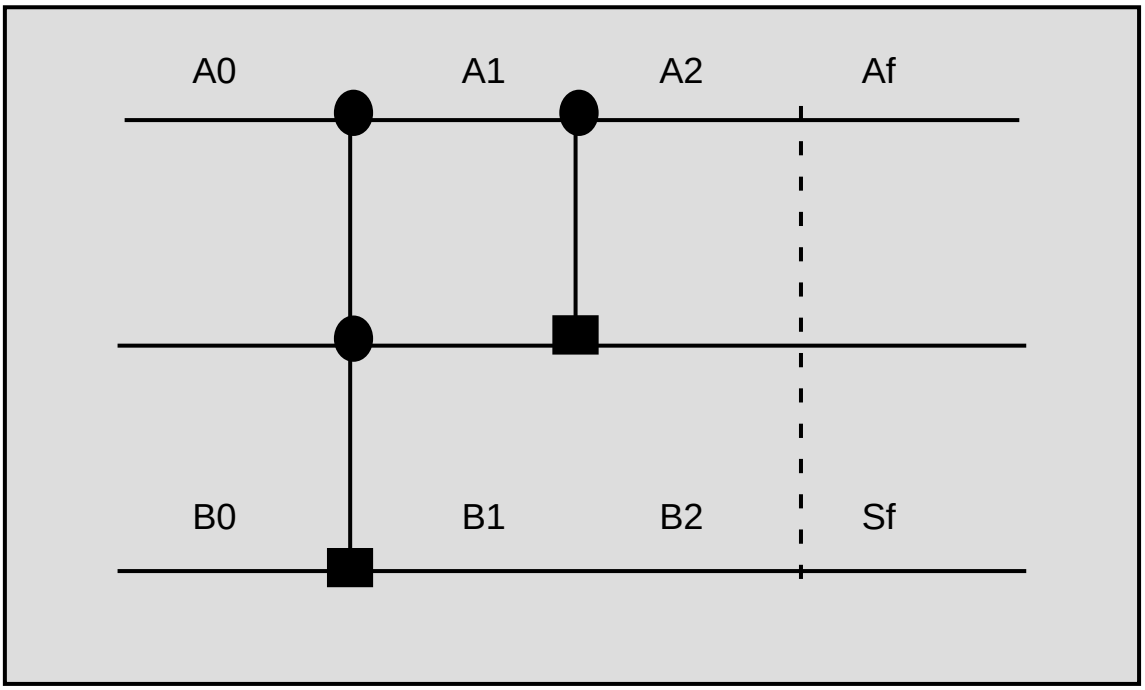
Arquitectura cuántica de un intercambio.

Evolución de estados en un intercambio construido combinando 3 puertas ⟨CN⟩							
|A0⟩	**|B0⟩**	**|A1⟩**	**|B1⟩**	**|A2⟩**	**|B2⟩**	**|Af⟩**	**|Bf⟩**
0	0	0	0	0	0	0	0
0	1	0	1	1	1	1	0
1	0	1	1	0	1	0	1
1	1	1	0	1	0	1	1

Para terminar con las puertas, simplemente recordar que la puerta de Fredkin realiza un intercambio controlado de las líneas de la entrada, tal y como se ilustra en la tabla siguiente:

Tabla de verdad de un intercambio controlado o puerta de Fredkin					
|A0⟩	**|B0⟩**	**|C0⟩**	**|Af⟩**	**|Bf⟩**	**|Cf⟩**
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
0	1	1	0	1	1
1	0	1	1	1	0
1	1	0	1	0	1
1	1	1	1	1	1

El mismo tratamiento puede darse a cualquier operación más compleja. Por ejemplo, en el caso del semisumador de la figura, habría que considerar la evolución de los estados de cada línea, en cada paso del proceso de suma tal y como se describe en la tabla correspondiente.



Y otra vez el semisumador.

|A0⟩	|B0⟩	|C0⟩	|A1⟩	|B1⟩	|C1⟩	|Af⟩	|Sf⟩	|Kf⟩	(|A0⟩ + |B0⟩)		(|Sf⟩ + |Kf⟩)	
0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	1	0	0	1	0	0	1	1	0
1	0	0	1	0	0	1	1	0	1	0	1	0
1	1	0	1	1	1	1	0	1	1	1	0	1

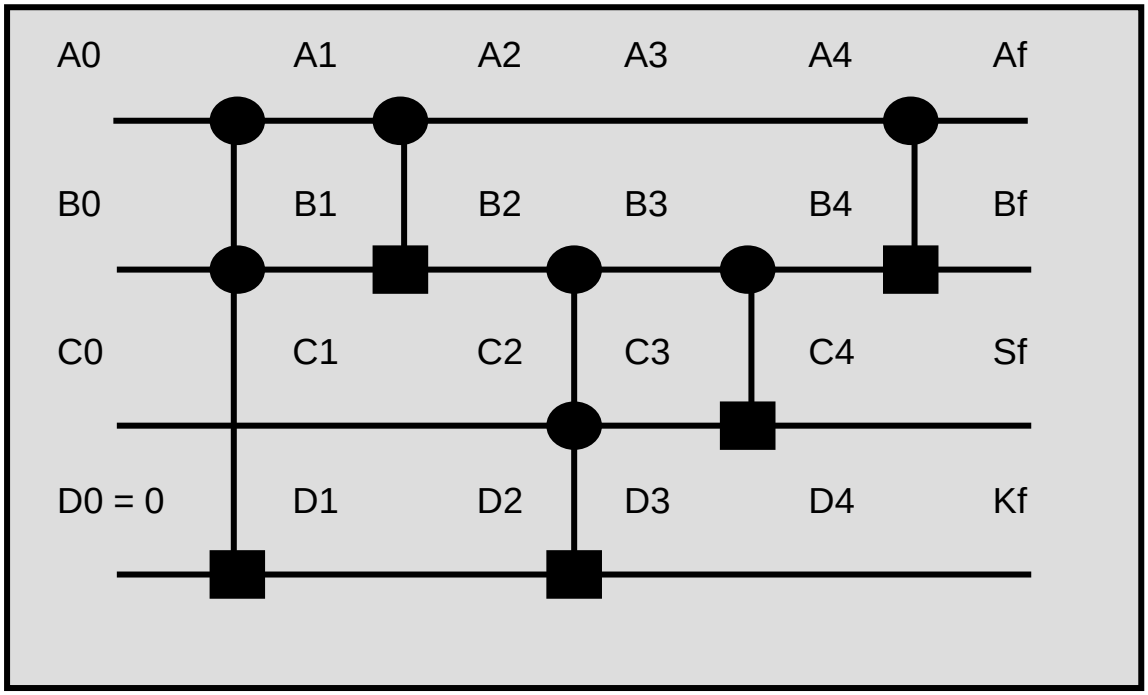
Evolución de los estados de cada línea en cada paso del proceso de suma de un semisumador.

En cuanto al sumador completo la solución pasa por seguir los pasos indicados en la figura, que se corresponden con la estructura lógica que se muestra a continuación, y cuyo resultado final aparece representado en la tabla.

$$\langle \sum_c [A_0, B_0, C_0, D_0 = 0] \rangle = [A_f, B_f, S_f, K_f]$$

Operación efectuada:

$$\langle \boldsymbol{\sum_c} \rangle = |CCN(D_{a,b})||CN(B_a)||CCN(D_{b,c})||CN(C_b)||CN(B_a)|$$



El sumador completo de números de dos bits.

|A0⟩ + |B0⟩ = |Sf⟩ + |Kf⟩				**Comentarios sobre arrastres previos**
0	0	0	0	Sin acarreo previo
0	0	1	0	Con acarreo previo
0	1	1	0	Sin acarreo previo
0	1	0	1	Con acarreo previo
1	0	1	0	Sin acarreo previo
1	0	0	1	Con acarreo previo
1	1	0	1	Sin acarreo previo
1	1	1	1	Con acarreo previo

Tabla del sumador completo de números de dos bits.

Diseño del Computador Cuántico

Vamos a entrar ahora en el meollo de la cuestión: el diseño de un operador que nos sirva para realizar computaciones al más puro estilo cuántico.

Si nos fijamos, hasta ahora sólo hemos necesitado cuatro átomos [a, b, c, d] para representar casi cualquier operación que hemos necesitado. Además, cada átomo podía ser un bit $|0\rangle$ o un bit $|1\rangle$. Considerado como un todo, nuestro sistema cuántico de cuatro átomos estará, en un instante dado, en el estado: $|a, b, c, d\rangle$. Si llamamos **M** a la matriz de la unidad o estructura lógica que genera la transformación, entonces podemos considerar lo siguiente:

$M\ |a, b, c, d\ \rangle = |a', b', c', d'\rangle$

$|\Psi in\rangle = |a, b, c, d\ \rangle \equiv$ Estado del sistema definido por la entrada

$|\Psi out\rangle = |a', b', c', d'\rangle \equiv$ Salida del sistema

Entonces: $|\Psi out\rangle = M\ |\Psi in\rangle$

Vamos a ilustrar estas ideas con un ejemplo. Supongamos que queremos realizar una suma completa sobre un sistema cuyo estado inicial es:

$$|\Psi in\rangle = |1, 0, 1, 0\rangle$$

Evidentemente, el operador **M**, tal y como lo hemos denotado antes, será $\langle\sum_c\rangle$, y –como se puede comprobar sin más que construir la tabla de verdad del sumador completo–, el resultado será:

$$|\Psi out\ \rangle = M\ |\ \Psi in\ \rangle = \langle\sum_c\rangle\ |1, 0, 1, 0\rangle = |1, 0, 0, 1\rangle$$

donde –como ya hemos dicho–

$$\langle \sum_c \rangle = |CCN\ (D_{a,b})||CN(B_a)||CCN(D_{b,c})||CN(C_b)||CN(B_a)|$$

Aquí hemos representado las operaciones reversibles según el orden de aplicación convencional de las operaciones primitivas sucesivas.

Si utilizamos la notación de Feynman podemos escribir $\langle \sum_c \rangle$ como un producto de matrices, en donde el orden de aplicación es de derecha a izquierda:

$$\langle \sum_c \rangle = A_{a,b}\, A_{b,c}\, A_{bc,d}\, A_{a,b}\, A_{ab,d}$$

Para aclarar un poco las cosas, y aunque no es necesario:

- $A_{a,b} = \langle CN(B_a) \rangle$
- $A_{b,c} = \langle CN(C_b) \rangle$
- $A_{bc,d} = \langle CCN(D_{b,c}) \rangle$
- $A_{ab,d} = \langle CCN(D_{a,b}) \rangle$

La formulación general del problema es la siguiente: Sea $\{A_1, A_2, A_3, \ldots A_k\}$ la sucesión o secuencia de operaciones requerida para que una determinada unidad lógica compleja opere sobre |n| líneas. Necesitamos una matriz $\mathbf{M_n}$ de $2^n \times 2^n$ que podemos construir como:

$$\mathbf{M_n} \sim A_k \ldots A_3\, A_2\, A_1$$

donde cada A_i de este producto es una matriz sencilla, lo que equivale a una operación elemental. Pero el problema ahora es: ¿cómo generamos físicamente $\mathbf{M_n}$ si sabemos construir los elementos más sencillos?

Al respecto, la mecánica cuántica predice que el estado de la salida de un sistema en un instante cualquiera $|\tau|$ es:

$$|\Psi out\rangle = e^{\,i\tau H}\, |\Psi in\rangle = \exp(i\tau H)\, |\Psi in\rangle$$

Donde:

- τ es el tiempo
- H es el Hamiltoniano
- $|\Psi in\rangle$ es el estado de la entrada
- $|\Psi out\rangle$ es el estado de la salida

Evidentemente:

$$\mathbf{M_n} = e^{i\tau H} = \exp(i\tau H)$$

Tenemos que encontrar el Hamiltoniano de forma que:

- H / dado $\tau \rightarrow \mathbf{M_n} = e^{i\tau H} = \exp(i\tau H)$
- $\mathbf{M_n}$ es un producto de matrices no conmutativas.

Pero hacer lo anterior a partir de alguna propiedad sencilla de las propias matrices es una tarea formidable. Afortunadamente podemos resolver la cuestión, al menos en parte, considerando que:

- Para un τ dado : $e^{i\tau H} = \exp(i\tau H) \approx 1 + i\tau H - (H^2\tau^2)/2 \ldots$
- El Hamiltoniano opera un número arbitrario de veces.
- El estado global se obtiene por superposición.

Y para realizar la composición de matrices A_i hacemos lo siguiente:

Sean los |n| átomos del registro. Añadimos un conjunto nuevo de |k+1| átomos que configuran lo que vamos a llamar el contador de posiciones del programa. Denotamos como $\langle q_i \rangle$ al operador de aniquilación de la posición |i| y como $\langle q^*_i \rangle$ al operador de creación de la posición |i|, de tal forma que ambos $\langle q_i \rangle$ y $\langle q^*_i \rangle$ operan desde |i = 0| hasta |i = k|. Necesitamos ahora un electrón cambiando continuamente de una posición a otra. Así, si en un τ dado una posición está vacía el estado de esa posición es $|0\rangle$, y si en un τ dado una posición está ocupada el estado de esa posición es $|1\rangle$. Con este planteamiento Feynman propone como Hamiltoniano:

$$\mathrm{H} = \sum_{i=0}^{k-1} q_{i+1}^{*} q_i A_{i+1} + complejo\_conjugado$$

$$\mathrm{H} = q_1^{*} q_0 A_1 + q_2^{*} q_1 A_2 + q_3^{*} q_2 A_3 + \ldots + q_0^{*} q_1 A_1^{*} + q_1^{*} q_2 A_2^{*} + q_2^{*} q_3 A_3^{*} + \ldots$$

Vamos a ver con algo de detalle el funcionamiento de todo este follón:

Si todas las posiciones del programa están libres, entonces todos los átomos del programa están en el estado $|0\rangle$, por lo tanto no hay cambios ya que cada término del Hamiltoniano comienza con un operador de aniquilación. Esto significa que la expresión para H sólo es cierta (H es ν) cuando una y sólo una de las posiciones del programa está ocupada.

$$\mathrm{H\ es\ } \nu \leftrightarrow |\Psi i\rangle = |1\rangle \ \wedge |\Psi j\rangle = |0\rangle : \ \forall i \ \forall j \neq i$$

Como consecuencia de lo anterior el número de posiciones del programa en estado $|1\rangle$ es siempre el mismo. Además, durante el proceso de cómputo sólo puede ocurrir que no haya posiciones ocupadas –en cuyo caso no pasa nada–, o que sólo haya una posición ocupada –en cuyo caso se realiza una computación elemental–. Por otra parte, durante un proceso normal de cómputo, dos o más posiciones de programa no pueden estar ocupadas simultáneamente.

Para tratar de entender qué es lo que está pasando en nuestra infernal computadora cuántica consideremos lo siguiente: Sea un sistema en un estado inicial $|\Psi 0\rangle$…

1. Cuando $|i = 0| \rightarrow$ Posición (Pos) $= 0 \rightarrow |Pos = 0\rangle = |1\rangle$
2. Dejamos transcurrir un tiempo arbitrario $|\tau|$
3. Observamos que $|Pos = k\rangle = |1\rangle$
4. Observamos que $|Pos \neq k\rangle = |0\rangle$
5. El registro $|n|$ ha sido multiplicado por la matriz : $\mathbf{M} = A_k \ldots A_3\ A_2\ A_1$
6. La operación ha sido realizada.

Qué… ¿todavía nada? Pues vamos a seguir intentándolo yéndonos un poco más al detalle.

1. Sea un registro de tres átomos: $|n| = 3$

2. Sean 4 operaciones elementales: $|k| = 4$

3. Número de posiciones del registro contador de programa: $|k + 1| = 5$

4. $k = 0 : k = 1 : k = 2 : k = 3 : k = 4$

5. $|Pos\rangle = |1,0,0,0,0\rangle : |\Psi 0\rangle \rightarrow |\Psi 1\rangle = A1\ |\Psi 0\rangle : |Pos\rangle = |0,1,0,0,0\rangle$

6. $|Pos\rangle = |0,1,0,0,0\rangle : |\Psi 1\rangle \rightarrow |\Psi 2\rangle = A2\ |\Psi 1\rangle : |Pos\rangle = |0,0,1,0,0\rangle$

7. $|Pos\rangle = |0,0,1,0,0\rangle : |\Psi 2\rangle \rightarrow |\Psi 3\rangle = A3\ |\Psi 2\rangle : |Pos\rangle = |0,0,0,1,0\rangle$

8. $|Pos\rangle = |0,0,0,1,0\rangle : |\Psi 3\rangle \rightarrow |\Psi 4\rangle = A4\ |\Psi 3\rangle : |Pos\rangle = |0,0,0,0,1\rangle$

Explicamos el ejemplo:

Al principio, el sistema de $|n|$ átomos está en el estado $|\Psi 0\rangle$ y la posición $|0|$ del registro contador está ocupada. En estas condiciones, el término de H que puede operar es $|q_1{}^*q_0A_1|$. La situación es:

- $q_0 : |pos\ (0)\rangle \rightarrow$ pos (desocupada)
- $q_1{}^* : |pos\ (0)\rangle \rightarrow$ pos (ocupada)
- $q_1{}^*q_0 : |1\rangle_{pos\ (0) \rightarrow pos\ (1)}$

Ahora multiplica la matriz A_1, que opera sobre los |n| átomos. No obstante, es conveniente recordar que si **a** y **a*** son, respectivamente operadores de aniquilación, el resultado de sus acciones sobre los estados posibles es el siguiente:

- Si **a** es un operador de aniquilación

$$\mathbf{a}: \quad |1\rangle \rightarrow |0\rangle$$
$$\mathbf{a}: \quad |0\rangle \rightarrow 0$$

- Si **a*** es un operador de creación

$$\mathbf{a}^*: \quad |0\rangle \rightarrow |1\rangle$$
$$\mathbf{a}^*: \quad |1\rangle \rightarrow 0$$

Por lo tanto q_0 aniquila la posición |0| y q_1^* crea en la posición |1|. Ahora se puede ver con claridad que:

- Si q_{i+1} es |aniquilación|, entonces para la posición $|i+1| \Rightarrow |1\rangle_{i+1} \rightarrow |0\rangle_{i+1}$
- Si q_i^* es |creación|, entonces para la posición $|i| \Rightarrow |0\rangle_i \rightarrow |1\rangle_i$

Describimos a continuación el funcionamiento del programa, para lo cual establecemos ciclos.

Primer Ciclo

- Registro contador de posiciones de programa

$|i|=0 : |Pos\ (i)\rangle = |1\rangle : |Pos\ (\ i+1 \rightarrow k\ )\rangle = |0\rangle$

$|Pos0\rangle = |1, 0, 0,\ldots,0\rangle$

- Registro de |n| átomos en su estado inicial

$|\Psi 0\rangle = |x_1, x_2, \ldots, x_n\rangle_0 : x_i = 0 , 1$

- Opera el primer término de H : $q_1 * q_0 A_1$

A_1 opera sobre el registro de |n| átomos: $|\Psi 1\rangle = A_1 |\Psi 0\rangle$

$q_1 * q_0 : |1, 0, 0, \ldots, 0\rangle \rightarrow |0, 1, 0, \ldots, 0\rangle$

Si A_1 quisiera volver a operar no podría porque $|Pos\ (0)\rangle = |0\rangle$

Segundo Ciclo

- Registro contador de posiciones de programa

$|i|=1 : |Pos\ (i)\rangle = |1\rangle : |Pos\ (j = 0 \rightarrow k , j \neq i)\rangle = |0\rangle$

$|Pos1\rangle = |0, 1, 0, \ldots, 0\rangle$

- Registro de |n| átomos en su nuevo estado

$|\Psi 1\rangle = |x_1, x_2, \ldots, x_n\rangle_1 : x_i = 0 , 1$

- Término de H que opera: $q_2 * q_1 A_2$

A2 opera sobre el registro de |n| átomos: $|\Psi 2\rangle = A_2 A_1 |\Psi 0\rangle$

$q_2 * q_1 : |0, 1, 0, \ldots, 0\rangle \rightarrow |0, 0, 1, \ldots, 0\rangle$

Se sigue moviendo el cursor y las operaciones elementales primitivas son aplicadas en el orden correcto y se va realizando la operación compleja **M.**

En todo este lío hay un conjunto de restricciones que siempre tenemos que considerar. Por ejemplo, un Hamiltoniano |H| tiene que ser hermítico por lo que los complejos conjugados tienen que estar presentes. Además, a una posición |j| se puede llegar desde |j − 1| o desde |j + 1|, pero el resultado tiene que ser el mismo

venga de donde venga… ¿se cumplen estas restricciones con nuestro diseño? Analicemos lo que pasa.

Sea $|Pos\ (1)\rangle = |1\rangle$ y ya ha operado $A_2A_1|Registro(n)\rangle$. Si ahora actúa $q_2{}^*q_1$, se movería el cursor a $|Pos\ (2)\rangle = |1\rangle$ lo se traduciría en $|\Psi 012\rangle = A3\ A2\ A1|\Psi 0\rangle$. Por el contrario, si actuase $q_1{}^*q_2$ el cursor se movería a $|Pos\ (0)\rangle = |1\rangle$ lo que se traduciría en el nuevo estado $|\Psi 010\rangle = A_2{}^*\ A_2\ A_1|\Psi 0\rangle$. Pero sabemos ya que $A_2{}^*\ A_2 = 1$ por lo cual –como no podía ser de otra manera– $|\Psi 010\rangle = A_1|\Psi 0\rangle$.

El resultado neto es que el estado del registro |n| depende de la posición del cursor.

Configuración del Computador Cuántico

Para poner en marcha nuestro computador cuántico podemos seguir el siguiente protocolo:

1. Configurar la entrada en el registro de |n| átomos.
2. Poner el cursor en la posición |0| : $|Pos\ (0)\rangle = |1\rangle$
3. Dejar que el sistema evolucione.
4. Observar continuamente (e.g.: por dispersión electrónica) la posición del cursor.
5. Cuando $|Pos\ (Final\rangle = |1\rangle$ hacer $|Pos\ (Final\rangle = |0\rangle$
6. Al tener el cómputo interrumpido ya podemos medir el registro |n|

El principal problema que vamos a tener, y que todavía dista mucho de estar resuelto es el de la interacción con el mundo exterior para configurar las entradas y leer la salida. De todas formas se puede demostrar matemáticamente que el comportamiento del cursor es análogo al de un conjunto de ondas de propagación de electrones fuertemente ligados, o al de un conjunto de ondas de espín en una dimensión, lo cual nos puede dar alguna pista.

Para facilitar la puesta en marcha, y la parada, de nuestra computadora cuántica supongamos que, además de las posiciones internas usadas en la computación, creamos una nueva línea de posiciones –muchas antes y muchas después– tal y como se muestra en la tabla.

-	Por aquí introducimos					Por aquí procesamos					Por aquí extraemos				
Cursor ⟨i⟩	–z	–y	–x	–t	…	0	1	2	…	k	k+1	k+2	k+3	k+4	…
Operación	×1	×1	×1	×1	×1	A_1	A_2	A_3	…	A_{k+1}	×1	×1	×1	×1	×1

Capturando el cursor en la computadora cuántica de Feynman.

De este modo puede considerarse como si tuviéramos valores del índice |i| de q_i tal que:

$$(i < 0) \wedge (i > k) \,/\, A_i \rightarrow (\times 1)$$

El funcionamiento de este engendro, en el que tenemos una cadena de espín mucho más larga, es el siguiente:

1. Antes de computar (i < 0) y la computadora no hace nada:

$$|\Psi i\rangle = |\Psi In\rangle \;\forall\; i < 0$$

2. Después de computar (i > k) y la computadora no hace nada:

$$|\Psi i\rangle = |\Psi Out\rangle \;\forall\; i < 0$$

3. Empezaremos con |i = 0| o con | i | en otras posiciones $\leftrightarrow$ i < 0

4. Terminaremos con |i = k| o con | i | en otras posiciones $\leftrightarrow$ i > k

Y así podemos capturar al cursor más fácilmente.

--..--

ALGORITMOS CUÁNTICOS RELEVANTES

Un algoritmo cuántico es un algoritmo que se ejecuta en un modelo realista de computación cuántica. La teoría de la complejidad computacional le asigna la clase BQP a los algoritmos que pueden ser resueltos en un computador cuántico en tiempo polinómico con un margen de error promedio inferior a 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.

Entre los algoritmo cuánticos destacables encontramos los siguientes: El algoritmo de Deutsch-Jozsa fue propuesto por David Deutsch y Richard Jozsa en 1992 y fue mejorado posteriormente por Richard Cleve, Artur Ekert, Chiara Macchiavello, y Michele Mosca en 1998.

David E. Deutsch (1953) es físico por la Universidad de Oxford, miembro de la Royal Society, y profesor visitante en el "Department of Atomic and Laser Physics" del "Centre for Quantum Computation", en el Clarendon Laboratory, de Oxford. Fue pionero en el campo de la computación cuántica, al ser el primero en formular un algoritmo cuántico, y es uno de los formuladores *de la teoría de los universos paralelos dentro de la mecánica cuántica*.

El propósito del algoritmo de Deutsch es determinar si una función de tipo caja negra:

$$f : \{0, 1\}^n \rightarrow \{0, 1\}$$

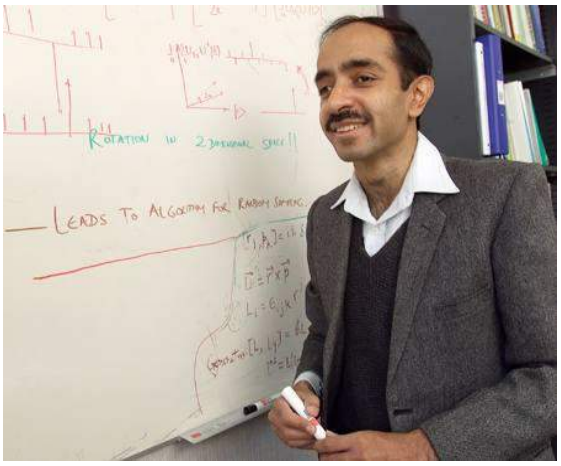
es «constante» o «balanceada». Esto es, dada una función que para una entrada de 'n' bits da un sólo bit de salida, determinar si la salida es independiente de la entrada, o si para la mitad de las entradas es 0 y para la otra mitad es 1. El

planteamiento del problema excluye todas las otras posibles funciones. El algoritmo no tiene apenas utilidad práctica, pero es uno de los primeros ejemplos de un algoritmo cuántico que se ha demostrado que es exponencialmente más rápido que cualquier posible algoritmo clásico determinista.

Otro algoritmo que estudiaremos después de formular la Transformada Cuántica de Fourier, que es necesaria para su desarrollo, será el algoritmo de Shor. Dicho algoritmo es el responsable de buena parte de la atención que se le ha dedicado a la computación cuántica por su relación con problemas de importancia fundamental en criptografía.

A continuación analizaremos el algoritmo de Simon, propuesto por Daniel Simon en 1994, y mediante el cual se trata de encontrar el periodo de una función vectorial booleana.

Finalizaremos nuestra presentación de algoritmos cuánticos comentando el algoritmo de Grover, publicado por Lov Grover en 1996, y con el que se pudo demostrar que un problema de utilidad práctica podía ser resuelto más rápidamente que el mejor algoritmo clásico posible. El algoritmo realiza una búsqueda en una base de datos desordenada con N entradas .



Lov Grover. científico informático indio-americano, creador del algoritmo de búsqueda de Grover en bases de datos usado en la computación cuántica. Obtuvo su licenciatura en el Instituto Indio de Tecnología de Delhi. Fue profesor en la Universidad de Cornell. Actualmente es investigador en los Laboratorios Bell de Nueva Jersey.

No se estudiarán, sin embargo, otros algoritmos como por ejemplo el desarrollo de la primera corrección de errores cuántica, propuesta también por

Peter Shor en 1995, y que fue el primer paso hacia la computación cuántica a prueba de errores... ¡Algo hay que dejar para que el lector interesado se rompa la cabeza!

Algoritmos de Deutsch y de Deutsch-Jozsa

El algoritmo de Deutsch-Jozsa fue uno de los primeros algoritmos diseñados para ejecutar sobre un computador cuántico y tiene el potencial de ser más eficiente que los algoritmos clásicos al aprovechar el paralelismo inherente de los estados de superposición cuánticos.

En el problema de Deutsch-Jozsa nos dan una función cuántica (que para nosotros es una caja negra) $f(x_1, x_2,..., x_n)$ que toma n bits de entrada $x_1, x_2,..., x_n$ y devuelve un valor binario $f(x_1, x_2,..., x_n)$. Sabemos que la función es constante (0 en todas las entradas o 1 en todas las entradas) o balanceada (devuelve 1 para la mitad de las entradas y 0 para la otra mitad). El problema es entonces determinar cómo es la función (constante o balanceada) aplicando entradas a la caja negra y observando su salida.

Analizaremos primero una versión del algoritmo para una función $f(x)$ de una sola entrada. Primero trataremos el problema desde una perspectiva clásica, y luego haremos un planteamiento cuántico del mismo.

Desde un punto de vista clásico, existen cuatro funciones posibles que satisfacen los requisitos del problema. Así:

- Si la entrada es '0', la salida puede ser:

  1. $f_1(0) = 0$; $f_2(0) = 0$

  2. $f_3(0) = 1$; $f_4(0) = 1$

- Si la entrada es '1', la salida puede ser:

  3. $f_1(1) = 0$; $f_2(1) = 1$

4. $f_3(1) = 0$; $f_4(1) = 1$

De acuerdo con lo establecido, f_1 y f_4 son constantes (ie., f_1 siempre devuelve '0' y f_4 siempre devuelve '1') mientras que f_2 y f_3 son balanceadas (ie., f_2 y f_3 devuelven '0' o devuelven '1' dependiendo de la entrada). Podemos construir ahora unas funciones:

$$x = f(0) : y = f(1)$$

y una máquina de Turing:

$$M(x, y) = 1 \quad \leftrightarrow \quad x = y$$

$$M(x, y) = 0 \quad \leftrightarrow \quad x \neq y$$

Sin embargo, esta máquina de Turing requiere explícitamente que evaluemos la función 'f' dos veces. La pregunta ahora es: ¿No podríamos resolver el problema evaluando la función una sola vez? Esto equivale a decir que necesitamos una máquina de Turing (M') tal que:

$$M'(x) = 1 \quad \leftrightarrow \quad x = y$$

$$M'(x) = 0 \quad \leftrightarrow \quad x \neq y$$

Sin embargo, esta máquina de Turing es imposible de construir siguiendo un planteamiento clásico, porque 'y' no es una entrada del sistema. Vamos a tratar de resolver el problema cuánticamente. Para ello consideremos los qubits $|0\rangle$ y $|1\rangle$ sobre los cuales aplicamos una transformación de Hadamard, de forma que:

$$U_{hadamard}|0\rangle \rightarrow \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right) : U_{hadamard}|1\rangle \rightarrow \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right)$$

Llamaremos:

$$|x\rangle = (|0\rangle + |1\rangle) : |y\rangle = (|0\rangle - |1\rangle)$$

en donde para mayor claridad hemos prescindido de los factores de normalización.

El objetivo es construir una puerta cuántica de dos entradas y de dos salidas que realice la transformación:

$$U\,|x, y\rangle \rightarrow |x, y \oplus f(x)\rangle$$

Veamos por qué:

- Primero configuramos la entrada:

  $|\psi_{Entrada}\rangle = |x, y\rangle = |x\rangle \otimes |y\rangle = (|0, 0\rangle - |0, 1\rangle + |1, 0\rangle - |1, 1\rangle)$

  que es un estado entrelazado y superpuesto que permite evaluar la función 'f' una sola vez.

- Aplicamos la transformación propuesta sobre la entrada con lo que obtenemos la salida:

  $$|\psi_{Salida}\rangle = U\,|\psi_{Entrada}\rangle =$$

  $$= (|0, 0 \oplus f(0)\rangle - |0, 1 \oplus f(0)\rangle + |1, 0 \oplus f(1)\rangle - |1, 1 \oplus f(1)\rangle)$$

- Analizamos ahora cada uno de los términos de $= |\psi_{Salida}\rangle$:

  a. $|0, 0 \oplus f(0)\rangle$

  - Si $f(0) = 0 \rightarrow |0, 0 \oplus f(0)\rangle = |0, 0 \oplus 0\rangle = |0, 0\rangle = |0, f(0)\rangle$

  - Si $f(0) = 1 \rightarrow |0, 0 \oplus f(0)\rangle = |0, 0 \oplus 1\rangle = |0, 1\rangle = |0, f(0)\rangle$

  b. $|0, 1 \oplus f(0)\rangle$

  - Si $f(0) = 0 \rightarrow |0, 1 \oplus f(0)\rangle = |0, 1 \oplus 0\rangle = |0, 1\rangle = |0, \neg f(0)\rangle$

  - Si $f(0) = 1 \rightarrow |0, 1 \oplus f(0)\rangle = |0, 1 \oplus 1\rangle = |0, 0\rangle = |0, \neg f(0)\rangle$

  c. $|1, 0 \oplus f(1)\rangle$

- Si $f(1) = 0 \rightarrow |1, 0 \oplus f(1)\rangle = |1, 0 \oplus 0\rangle = |1, 0\rangle = |1,\ f(1)\rangle$

- Si $f(1) = 1 \rightarrow |1, 0 \oplus f(1)\rangle = |1, 0 \oplus 1\rangle = |1, 1\rangle = |1,\ f(1)\rangle$

d. $|1, 1 \oplus f(1)\rangle$

- Si $f(1) = 0 \rightarrow |1, 1 \oplus f(1)\rangle = |1, 1 \oplus 0\rangle = |1, 1\rangle = |1, \neg f(1)\rangle$

- Si $f(1) = 1 \rightarrow |1, 1 \oplus f(1)\rangle = |1, 1 \oplus 1\rangle = |1, 0\rangle = |1, \neg f(1)\rangle$

De este modo:

$$U\,|\psi_{Entrada}\rangle = U\,(|0, 0\rangle - |0, 1\rangle + |1, 0\rangle - |1, 1\rangle) =$$

$$= (|0, 0 \oplus f(0)\rangle - |0, 1 \oplus f(0)\rangle + |1, 0 \oplus f(1)\rangle - |1, 1 \oplus f(1)\rangle) =$$

$$= (|0, f(0)\rangle - |0, \neg f(0)\rangle + |1, f(1)\rangle - |1, \neg f(1)\rangle) =$$

$$= \{\, |0\rangle \otimes (\, |f(0)\rangle - |\neg f(0)\rangle) + |1\rangle \otimes (\, |f(1)\rangle - |\neg f(1)\rangle)\,\} = |\psi_{Salida}\rangle$$

Aparentemente no hemos hecho gran cosa, puesto que seguimos teniendo que evaluar 'f(0)' y 'f(1)'. Pero recordemos que:

- Si 'f' es constante $\rightarrow f(0) = f(1)$

$$|\psi_{Salida}\rangle = \{\, |0\rangle \otimes (\, |f(0)\rangle - |\neg f(0)\rangle) + |1\rangle \otimes (\, |f(1)\rangle - |\neg f(1)\rangle)\,\} =$$

$$= \{\, |0\rangle \otimes (\, |f(0)\rangle - |\neg f(0)\rangle) + |1\rangle \otimes (\, |f(0)\rangle - |\neg f(0)\rangle)\,\} =$$

$$= (|0\rangle + |1\rangle) \otimes (|f(0)\rangle - |\neg f(0)\rangle)$$

- Si 'f' es balanceada $\rightarrow f(0) = \neg f(1)$

$$|\psi_{Salida}\rangle = \{\, |0\rangle \otimes (\, |f(0)\rangle - |\neg f(0)\rangle) + |1\rangle \otimes (\, |f(1)\rangle - |\neg f(1)\rangle)\,\} =$$

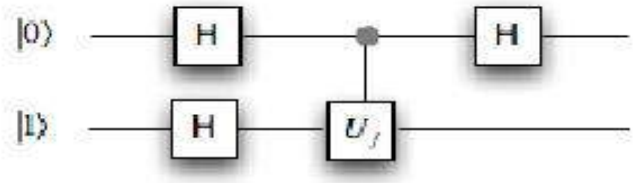
$$= \{\, |0\rangle \otimes (\, |f(0)\rangle - |\neg f(0)\rangle) + |1\rangle \otimes (\, |\neg f(0)\rangle - |\, f(0)\rangle)\,\} =$$

$$= \{ |0\rangle \otimes (|f(0)\rangle - |\neg f(0)\rangle) - |1\rangle \otimes (|f(0)\rangle - |\neg f(0)\rangle) \} =$$

$$= (|0\rangle - |1\rangle) \otimes (|f(0)\rangle - |\neg f(0)\rangle)$$

Por lo tanto, con solo evaluar 'f(0)' podemos saber si la función es constante o balanceada. Para ello hay que medir el primer qubit del estado de salida sobre la base $\{ |0\rangle + |1\rangle , |0\rangle - |1\rangle \}$. Si después de medir obtenemos $(|0\rangle + |1\rangle)$ la función es constante, y si obtenemos $(|0\rangle - |1\rangle)$ la función es balanceada.

Una forma de implementar este algoritmo es a través del circuito cuántico de la figura.



Circuito cuántico para el algoritmo de Deutsch

Las puertas de Hadamard, H, son importantes para el desarrollo de algoritmos cuánticos en general, y para el de Deutsch en particular. Interesa recordar que, como puerta reversible, la aplicación en serie de H sobre un estado cualquiera nos devuelve el estado inicial: $H^2 = I$. Sin embargo, su aplicación en paralelo nos (como en este caso), nos genera el producto de dos estados superpuestos. Por ejemplo:

$$(H \otimes H) |1\rangle |1\rangle = (H |1\rangle) (H |1\rangle = (1/\sqrt{2}) (|0\rangle - |1\rangle) (1/\sqrt{2}) (|0\rangle - |1\rangle) =$$

$$= (1/2) (|00\rangle - |01\rangle + |10\rangle - |11\rangle)$$

En este contexto, llamaremos Transformada de Hadamard a la aplicación de 'n' puertas de Hadamard en paralelo sobre n-qubits, y la denotaremos como $H^{\otimes n}$. Por ejemplo $H^{\otimes 3}$ sobre $|0\rangle\ |0\rangle\ |0\rangle$ habría que interpretarlo como sigue:

$$H^{\otimes 3}(|0\rangle\ |0\rangle\ |0\rangle) = (H \otimes H \otimes H)\ |000\rangle = (H\ |0\rangle)\ (H\ |0\rangle)\ (H\ |0\rangle) =$$

$$= (1/\sqrt{2})\ (|0\rangle + |1\rangle)\ (1/\sqrt{2})\ (|0\rangle + |1\rangle)\ (1/\sqrt{2})\ (|0\rangle + |1\rangle) =$$

$$(1/\sqrt{2^3})\ (|000\rangle + |001\rangle + |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle)$$

En general, si $|x\rangle$ es un estado tal que $\{x = 0, \ldots, 2^n - 1\}$, la aplicación de $H^{\otimes n}$ sobre un estado con 'n' entradas $|0\rangle$ se puede representar de la forma siguiente:

$$H^{\otimes n}|0\rangle^{\otimes n} = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle$$

De acuerdo con este planteamiento, los bloques H de la figura anterior son puertas de Hadamard cuya operación, como sabemos, es la siguiente:

$$U_{hadamard}|0\rangle \rightarrow \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right) : U_{hadamard}|1\rangle \rightarrow \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right)$$

Por otra parte, el bloque **U** realiza la transformación:

$$U\ |x_i\ ,\ y_i\rangle = |\ x_i\ ,\ y_i \oplus f(x_i)\rangle\ /\ x_i\ ,\ y_i \in \{0,\ 1\}$$

En el algoritmo que describiremos a continuación, haremos que la entrada $|x_i\rangle$ esté en un estado superpuesto conseguido a través de las puertas de Hadamard. Según se deduce de la figura, el procedimiento será:

$$|\psi_{salida}\rangle = (H \otimes I)\ U\ (H \otimes H)\ |0\rangle\ |1\rangle$$

El algoritmo implica los siguientes pasos:

- Definir un estado inicial $|\psi_{entrada}\rangle = |0\rangle|1\rangle$ y aplicar en paralelo las puertas de Hadamard para generar el producto de estados de dos superposiciones.
- Aplicar la transformación U al estado obtenido en el paso anterior.
- Sobre este nuevo resultado aplicar una puerta de Hadamard al primer qubit, dejando el segundo qubit sin modificar.

El proceso evoluciona del siguiente modo:

$$H|0\rangle = (1/\sqrt{2})(|0\rangle + |1\rangle) \quad : \quad H|1\rangle = (1/\sqrt{2})(|0\rangle - |1\rangle)$$

$$(H \otimes H)|0\rangle|1\rangle = (1/2)(|00\rangle - |01\rangle + |10\rangle - |11\rangle)$$

Ahora tenemos que aplicar U a este resultado:

$$U(1/2)(|00\rangle - |01\rangle + |10\rangle - |11\rangle) = (1/2)(U|00\rangle - U|01\rangle + U|10\rangle - U|11\rangle)$$

Antes de continuar con el desarrollo, conviene recordar que tanto f(0) como f(1) pueden valer 0 ó 1, y darse cuenta de que:

- Si $f(0) = 0 \rightarrow 0 \oplus f(0) = 0$: Si $f(1) = 0 \rightarrow 0 \oplus f(1) = 0$
- Si $f(0) = 1 \rightarrow 0 \oplus f(0) = 1$: Si $f(1) = 1 \rightarrow 0 \oplus f(1) = 1$
- Si $f(0) = 0 \rightarrow 1 \oplus f(0) = 1$: Si $f(1) = 0 \rightarrow 1 \oplus f(1) = 1$
- Si $f(0) = 1 \rightarrow 1 \oplus f(0) = 0$: Si $f(1) = 1 \rightarrow 1 \oplus f(1) = 0$

Aplicamos las transformaciones oportunas:

- $U|0,0\rangle = |0, 0 \oplus f(0)\rangle = |0, f(0)\rangle = (1 - f(0))|00\rangle + f(0)|01\rangle$
- $U|0,1\rangle = |0, 1 \oplus f(0)\rangle = |0, \neg f(0)\rangle = f(0)|00\rangle + (1 - f(0))|01\rangle$
- $U|1,0\rangle = |1, 0 \oplus f(1)\rangle = |1, f(1)\rangle = (1 - f(1))|10\rangle + f(1)|11\rangle$
- $U|1,1\rangle = |1, 1 \oplus f(1)\rangle = |1, \neg f(1)\rangle = f(1))|10\rangle + (1 - f(1))|11\rangle$

Desarrollamos ahora la expresión:

$$U\left(H\otimes H\right)|0\rangle|1\rangle=\frac{1}{2}\left(1-f(0)\right)|00\rangle+\frac{1}{2}f(0)|01\rangle-\frac{1}{2}f(0)|00\rangle-\frac{1}{2}\left(1-f(0)\right)|01\rangle+$$
$$+\frac{1}{2}\left(1-f(1)\right)|10\rangle+\frac{1}{2}f(1)|11\rangle-\frac{1}{2}f(1)|10\rangle-\frac{1}{2}\left(1-f(1)\right)|11\rangle$$
$$=\left(\frac{1}{2}-f(0)\right)|0\rangle|0\rangle-\left(\frac{1}{2}-f(0)\right)|0\rangle|1\rangle+$$
$$+\left(\frac{1}{2}-f(1)\right)|1\rangle|0\rangle-\left(\frac{1}{2}-f(1)\right)|1\rangle|1\rangle$$

Separando términos, la expresión puede representarse como sigue:

$$|\psi'\rangle=U\left(H\otimes H\right)|0\rangle|1\rangle=\left(\frac{1}{2}-f(0)\right)|0\rangle\left[|0\rangle-|1\rangle\right]+\left(\frac{1}{2}-f(1)\right)|1\rangle\left[|0\rangle-|1\rangle\right]=$$
$$=\left\{\left(\frac{1}{2}-f(0)\right)|0\rangle+\left(\frac{1}{2}-f(1)\right)|1\rangle\right\}\left\{|0\rangle-|1\rangle\right\}$$

Recordamos ahora que:

$$|\psi_{salida}\rangle = (H \otimes I)\ U\ (H \otimes H)\ |0\rangle\ |1\rangle = |\psi_{salida}\rangle = (H \otimes I)\ |\psi'\rangle$$

Esto significa que realizamos una transformación de Hadamard sobre el primer qubit de $|\psi'\rangle$ y dejamos el segundo qubit sin modificar. Por lo tanto, en nuestro caso:

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

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

$$(H\otimes I)|\psi'\rangle=\frac{1}{\sqrt{2}}\left\{\left(\frac{1}{2}-f(0)\right)\left(|0\rangle+|1\rangle\right)+\left(\frac{1}{2}-f(1)\right)\left(|0\rangle-|1\rangle\right)\right\}\left\{|0\rangle-|1\rangle\right\}=$$

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

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

Analizamos la salida:

- Si f(0) = f(1) $\rightarrow$ f(1) – f(0) = 0

  $|\psi_{salida}\rangle$ = (1 – f(0) – f(1)) $|0\rangle$ {(1/$\sqrt{2}$) ($|0\rangle$ – $|1\rangle$) = ± $|0\rangle$ {(1/$\sqrt{2}$) ($|0\rangle$ – $|1\rangle$)

- Si f(0) ≠ f(1) $\rightarrow$ 1 – f(0) – f(1) = 0

  $|\psi_{salida}\rangle$ = (f(1) – f(0)) $|1\rangle$ {(1/$\sqrt{2}$) ($|0\rangle$ – $|1\rangle$) = ± $|1\rangle$ {(1/$\sqrt{2}$) ($|0\rangle$ – $|1\rangle$)

Por lo tanto, con el dispositivo anterior y el algoritmo diseñado, si al medir el primer qubit obtenemos el valor 0, la función es constante. Por el contrario, si obtenemos un 1, la función es balanceada.

La computación cuántica permite resolver el problema de Deutsch ya que es capaz de evaluar simultáneamente f(0) y f(1). Esta posibilidad deriva del llamado 'paralelismo cuántico', que puede ser descrito convenientemente generalizando el problema de Deutsch, y es la base del algoritmo que resuelve el problema de Deutsch-Jozsa, ya enunciado. Al respecto, para generalizar el

problema de Deutsch, sea 'f' una función implementada por un circuito cuántico Uf tal que:

$$U_f |x, y\rangle \rightarrow |x, y \oplus f(x)\rangle$$

En este caso, para obtener el valor de f(x) hay evaluar:

$$U_f |x, 0\rangle \rightarrow |x, 0 \oplus f(x)\rangle = |x, f(x)\rangle$$

El aspecto más importante del proceso consiste en elegir un estado superpuesto inicial adecuado. En este contexto hay que recordar que un estado inicial superpuesto de n-qubits se podía representar mediante la ecuación:

$$\frac{1}{\sqrt{2^n}}\left(\left|0_1,...,0_{n-1},0_n\right\rangle + \left|0_1,...,0_{n-1},1_n\right\rangle + \left|0_1,...,1_{n-1},0_n\right\rangle + ... + \left|1_1,...,1_{n-1},1_n\right\rangle\right) = \frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}|x\rangle$$

$$U_f\left(\frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}|x,0\rangle\right) = \frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}U_f|x,0\rangle = \frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}|x, f(x)\rangle$$

En esto consiste el paralelismo cuántico, que supone que el operador U_f es aplicado simultáneamente a todos los vectores de la base que forman el estado superpuesto.

El paralelismo cuántico permite computar 2^n entradas para un estado formado por n-qubits 2n entradas; es decir, a partir de un crecimiento lineal del número de qubits se obtiene un crecimiento exponencial en el espacio de computación.

El algoritmo de Deutsch-Jozsa es una generalización del algoritmo de Deutsch que permite deducir si una función de n-qubits es constante o balanceada.

Antes de ver con detalle el procedimiento, consideremos la transformación U del algoritmo de Deutsch:

$$U\,|x, y\rangle = U\,|x\rangle\,|y\rangle = |x, y \oplus f(x)\rangle \;/\; x, y \in \{0, 1\}$$

Sea ahora:

$$|y\rangle = (1/\sqrt{2})\,(|0\rangle - |1\rangle)$$

y analicemos el comportamiento de U:

$$U|x\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)=\left(\frac{|x\rangle|0\oplus f(x)\rangle-|x\rangle|1\oplus f(x)\rangle}{\sqrt{2}}\right)=|x\rangle\left(\frac{|0\oplus f(x)\rangle-|1\oplus f(x)\rangle}{\sqrt{2}}\right)$$

Evaluemos la expresión que se muestra a continuación cuando f(x) = 0 y cuando f(x) = 1.

$$\left(\frac{|0\oplus f(x)\rangle-|1\oplus f(x)\rangle}{\sqrt{2}}\right)$$

- Si f(x) = 0, entonces:

$$\left(\frac{|0\oplus f(x)\rangle-|1\oplus f(x)\rangle}{\sqrt{2}}\right)=\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)\equiv(-1)^{f(0)}\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)$$

- Si f(x) = 1, entonces:

$$\left(\frac{|0\oplus f(x)\rangle-|1\oplus f(x)\rangle}{\sqrt{2}}\right)=-\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)\equiv(-1)^{f(1)}\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)$$

Por lo tanto, la expresión general puede escribirse como:

$$U|x\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)=|x\rangle(-1)^{f(x)}\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)=(-1)^{f(x)}|x\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)$$

Por otra parte, podemos considerar que el qubit de control |x⟩ se encuentra en un estado de superposición: $|x\rangle = a_0\,|0\rangle + a_1\,|1\rangle$. De este modo:

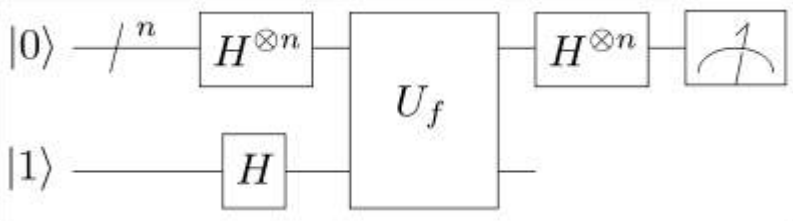
$$U|x\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)=U\left(a_0|0\rangle+a_1|1\rangle\right)\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)=$$

$$=\left((-1)^{f(0)}a_0|0\rangle+(-1)^{f(1)}a_1|1\rangle\right)\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)=\sum_{x=0}^{1}(-1)^{f(x)}a_x|x\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)$$

Por último, si la superposición de estados viene dada por la aplicación de 'n' puertas de Hadamard de la forma:

$$H^{\otimes n}|0\rangle^{\otimes n}=\frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}|x\rangle\rightarrow U\frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}|x\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)=\frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}(-1)^{f(x)}|x\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)$$

Dicho lo cual, estudiaremos el algoritmo general de Deutsch-Josza considerando el circuito cuántico de la figura.



Circuito cuántico del algoritmo de Deutsch-Josza.

El algoritmo empieza con un estado de 'n+1' qubits del tipo: $|0\rangle^{\otimes n}\ |1\rangle$. Para generar la entrada a la puerta U, realizamos una transformación de Hadamard sobre todos los qubits. Esto nos da:

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

Ahora aplicamos la transformación U $|x\rangle\ |y\rangle = |x\rangle\ |y \oplus f(x)\rangle$ a $|\psi_{enttrada}\rangle$. Obtenemos:

$$U|\psi_{entrada}\rangle = \frac{1}{\sqrt{2^{n+1}}}\sum_{x=0}^{2^n-1}|x\rangle\left(|\ f(x)\rangle - |1\oplus\ f(x)\rangle\right) = \frac{1}{\sqrt{2^{n+1}}}\sum_{x=0}^{2^n-1}(-1)^{f(x)}|x\rangle\left(|0\rangle - |1\rangle\right)$$

Tal y como se ilustra en la figura anterior, tenemos que realizar ahora una nueva transformación de Hadamard sobre los n-qubits correspondientes, lo que produce lo siguiente:

Si llamamos:

$$|\psi_{x-salida}\rangle = \frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}(-1)^{f(x)}|x\rangle \rightarrow H^{\otimes n}|\psi_{x-salida}\rangle = \frac{1}{2^n}\sum_{x=0}^{2^n-1}(-1)^{f(x)}\sum_{z=0}^{2^n-1}(-1)^{x\cdot z}|z\rangle =$$

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

Evidentemente:

$$x\cdot z = x_0z_0 \oplus x_1z_1 \oplus \ldots \oplus x_{n-1}z_{n-1}$$

Ahora podemos hacer dos cosas:

1. Observar la probabilidad, P, de la medida $|0\rangle^{\otimes n}$

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

P = 1 ↔ f(x) es constante

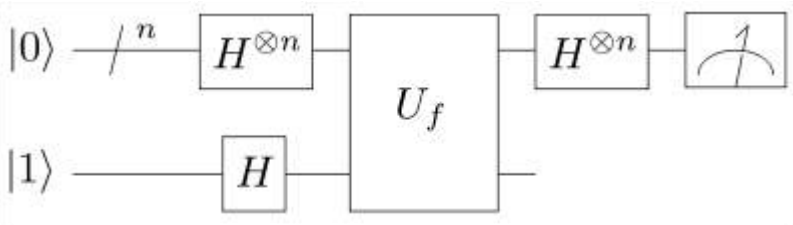
P = 0 ↔ f(x) es balanceada

2. Observar $|z\rangle$ después de la medida

Si f(x) es constante → z es cero (0,0,..., 0)

Si f(x) es balanceada → z es distinta de cero

Ilustraremos todo este lío con algún ejemplo. Consideremos el circuito cuántico definido para el algoritmo de Deutsch-Jozsa, sea f(x) = 1, y sea $|\psi_{inicial}\rangle = |001\rangle$. Procederemos paso a paso.



Otra vez el circuito cuántico del algoritmo de Deutsch-Josza.

$H|0\rangle = (1/\sqrt{2})(|0\rangle + |1\rangle)$: $H|1\rangle = (1/\sqrt{2})(|0\rangle - |1\rangle)$

Preparamos el sistema:

$$U|x, y\rangle = U|x\rangle|y\rangle = |x\rangle|y \oplus f(x)\rangle$$

$$H^{\otimes n}\,|0\rangle^{\otimes n} = H^{\otimes 2}\,|0\rangle^{\otimes 2} = H\,|0\rangle\,H\,|0\rangle = (1/\sqrt{2})\,[|0\rangle + |1\rangle]\,(1/\sqrt{2})\,[|0\rangle + |1\rangle] =$$

$$= (1/2)\,[|0\rangle\,|0\rangle + |0\rangle\,|1\rangle + |1\rangle\,|0\rangle + |1\rangle\,|1\rangle]$$

$$H\,|1\rangle = (1/\sqrt{2})\,(|0\rangle - |1\rangle)$$

La entrada a la puerta U es, por lo tanto:

$$|\psi_{\text{entrada-U}}\rangle = (1/2\sqrt{2})\,(|0\rangle|0\rangle|0\rangle - |0\rangle|0\rangle|1\rangle + |0\rangle|1\rangle|0\rangle - |0\rangle|1\rangle|1\rangle +$$

$$|1\rangle|0\rangle|0\rangle - |1\rangle|0\rangle|1\rangle + |1\rangle|1\rangle|0\rangle - |1\rangle|1\rangle|1\rangle)$$

Dado que, después de la transformación U, los dos primeros qubits no se modifican, y dado que la transformación implica que $U\,|x\rangle\,|y\rangle = |x\rangle\,|y \oplus f(x)\rangle$, y como hemos dicho que $f(x) = 1$, entonces el resultado de la transformación es el siguiente:

$$\begin{aligned} U\,|\psi_{\text{entrada-U}}\rangle = (1/2\sqrt{2}) \quad (& \quad |0\rangle|0\rangle|1\rangle - |0\rangle|0\rangle|0\rangle + \\ + & \quad |0\rangle|1\rangle|1\rangle - |0\rangle|1\rangle|0\rangle + \\ + & \quad |1\rangle|0\rangle|1\rangle - |1\rangle|0\rangle|0\rangle + \\ + & \quad |1\rangle|1\rangle|1\rangle - |1\rangle|1\rangle|0\rangle \quad) \end{aligned}$$

Factorizamos ahora la expresión considerando la salida de una puerta de Hadamard: $H\,|1\rangle = (1/\sqrt{2})\,(|0\rangle - |1\rangle)$. Así:

$$U\,|\psi_{\text{entrada-U}}\rangle = (1/2)\,\{\,(-|0\rangle|0\rangle - |0\rangle|1\rangle - |1\rangle|0\rangle - |1\rangle|1\rangle\,\}\,(1/\sqrt{2})\,\{\,|0\rangle - |1\rangle\,\}$$

Ahora podemos centrarnos en la parte de la expresión que, de acuerdo con la figura, vamos a medir -la llamaremos $|\psi'\rangle$- De este modo:

$$|\psi'\rangle = (1/2)\{\,(-|0\rangle|0\rangle - |0\rangle|1\rangle - |1\rangle|0\rangle - |1\rangle|1\rangle\,\} = -(1/2)\{\,(|0\rangle|0\rangle + |0\rangle|1\rangle + |1\rangle|0\rangle + |1\rangle|1\rangle\,\}$$

$$H^{\otimes 2}\,|\psi'\rangle^{\otimes 2} = (-1/2)(1/\sqrt{2})\{\,(|0\rangle + |1\rangle)\}\,(1/\sqrt{2})\{\,(|0\rangle + |1\rangle)\,\} +$$

$$+ \quad (-1/2)(1/\sqrt{2})\{(|0\rangle+|1\rangle)\}\,(1/\sqrt{2})\{(|0\rangle-|1\rangle)\} +$$

$$+ \quad (-1/2)(1/\sqrt{2})\{(|0\rangle-|1\rangle)\}\,(1/\sqrt{2})\{(|0\rangle+|1\rangle)\} +$$

$$+ \quad (-1/2)(1/\sqrt{2})\{(|0\rangle-|1\rangle)\}\,(1/\sqrt{2})\{(|0\rangle-|1\rangle)\} =$$

$$= \quad (-1/4)\{(|0\rangle+|1\rangle)\}\,\{(|0\rangle+|1\rangle)\} \quad +$$

$$= \quad (-1/4)\{(|0\rangle+|1\rangle)\}\,\{(|0\rangle-|1\rangle)\} \quad +$$

$$= \quad (-1/4)\{(|0\rangle-|1\rangle)\}\,\{(|0\rangle+|1\rangle)\} \quad +$$

$$= \quad (-1/4)\{(|0\rangle-|1\rangle)\}\,\{(|0\rangle-|1\rangle)\} \quad =$$

$$=(-1/4)|00\rangle+(-1/4)|01\rangle+(-1/4)|10\rangle+(-1/4)|11\rangle+$$

$$+(-1/4)|00\rangle+(-1/4)[-|01\rangle]+(-1/4)|10\rangle+(-1/4)[-|11\rangle]+$$

$$+(-1/4)|00\rangle+(-1/4)|01\rangle+(-1/4)[-|10\rangle]+(-1/4)[-|11\rangle]+$$

$$+(-1/4)|00\rangle+(-1/4)[-|01\rangle]+(-1/4)[-|10\rangle]+(-1/4)|11\rangle = -|00\rangle$$

Por lo tanto,

$$|\psi_{salida}\rangle = -|00\rangle\left(\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right)$$

Evidentemente, si medimos la salida $|x\rangle$, obtendremos el valor 'cero' con probabilidad total. Este resultado confirma que la función 'f (x)' es constante.

Transformada Cuántica de Fourier y Algoritmo de Shor

Hemos visto en los algoritmos estudiados que una de las armas más potentes en computación cuántica es la puerta de Hadamard. En realidad esta puerta es un caso especial de la llamada transformada cuántica de Fourier (QFT):

$$\sum_{x=0}^{N-1} f(x)|x\rangle \longrightarrow \sum_{y=0}^{N-1} \tilde{f}(y)|y\rangle$$

$$con \quad \tilde{f}(y) = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} e^{2\pi i x y/N} f(x)$$

Por tanto, según se deriva de las expresiones anteriores, y de acuerdo con lo ya mencionado, sobre los vectores de la base se aplica la transformación unitaria *NxN*:

$$U_{F_N}|x\rangle = \frac{1}{\sqrt{N}} \sum_{y=0}^{N-1} e^{2\pi i x y/N}|y\rangle$$

La QFT es la versión cuántica una transformada de Fourier discreta (DFT). En la versión clásica, en principio harían falta del orden de O(N^2) operaciones, pero en el caso de $N=2^n$, el orden O(2^{2n}) se puede reducir mediante la llamada transformada rápida de Fourier (FFT) a un orden de O($NlogN$)=O(n^{2n}). Aún así, sigue siendo un orden exponencial que, según veremos, mejora a cuadrático si empleamos la versión cuántica.

Asumiremos el caso binario, y sustituiremos en la última expresión la expansión binaria:

$$y = y_{n-1}y_{n-2}\cdots y_0 = y_{n-1}\cdot 2^{n-1} + \cdots + y_0 \cdot 2^0 \quad con \quad y_i \in \{0,1\}$$

obteniendo

$$U_{F_{2^n}} = \frac{1}{\sqrt{2^n}} \sum_{y_{n-1}=0}^{1} \cdots \sum_{y_0=0}^{1} \mathrm{e}^{2\pi i x \sum_{l=1}^{n} y_{n-l}/2^l} |y_{n-1} \cdots y_0\rangle$$

Ahora, si recordamos que la exponencial de una suma es el producto de las exponenciales resulta:

$$U_{F_{2^n}} = \frac{1}{\sqrt{2^n}} \bigotimes_{l=1}^{n} \left[\sum_{y_{n-l}=0}^{1} \mathrm{e}^{2\pi i x y_{n-l}/2^l} |y_{n-l}\rangle \right]$$

En consecuencia se puede ver que la QFT transforma la base computacional en otra base con vectores factorizables, es decir, sin entrelazamiento. A continuación, teniendo en cuenta la representación de las fracciones binarias:

$$\frac{x}{2} = x_{n-1}2^{n-2} + \cdots + x_1 + x_0 2^{-1} = x_{n-1} \cdots x_1 . x_0$$
$$\frac{x}{2^2} = x_{n-1}2^{n-3} + \cdots + x_2 + x_1 2^{-1} + x_0 2^{-2} = x_{n-1} \cdots x_2 . x_1 x_0$$

y despreciando la parte entera (que en la exponencial sólo formará unidades) se tiene:

$$U_{F_{2^n}} = \frac{1}{\sqrt{2^n}} (|0\rangle + \mathrm{e}^{2\pi i 0.x_0}|1\rangle) \otimes (|0\rangle + \mathrm{e}^{2\pi i 0.x_1 x_0}|1\rangle) \otimes \cdots \otimes (|0\rangle + \mathrm{e}^{2\pi i 0.x_{n-1} \cdots x_0}|1\rangle)$$

o de forma más compacta:

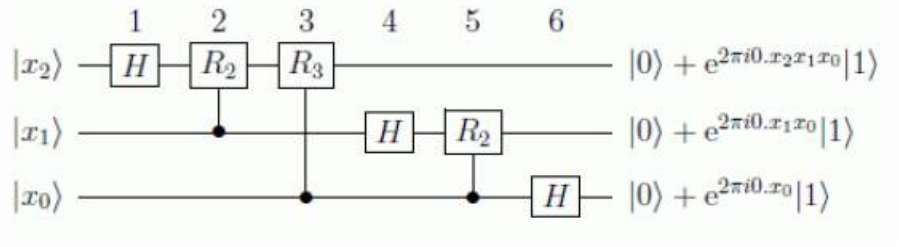
$$U_{F_{2^n}} = \frac{1}{\sqrt{2^n}} \bigotimes_{l=1}^{n} (|0\rangle + \mathrm{e}^{2\pi i 0.x_{l-1} \cdots x_0}|1\rangle)$$

Nótese que, como comentamos, la puerta de Hadamard no deja de ser una transformada de Fourier actuando sobre un solo qubit:

$$H|x\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + (-1)^x|1\rangle\right) = \frac{1}{\sqrt{2}}\left(|0\rangle + \mathrm{e}^{2\pi i x/2}|1\rangle\right) = \frac{1}{\sqrt{2}}\left(|0\rangle + \mathrm{e}^{2\pi i 0.x}|1\rangle\right)$$

Vamos a ver un circuito que implementa esta transformación. Sea el caso *n=3* con el estado de entrada:

$$|\psi_0\rangle = |x_2 x_1 x_0\rangle$$



En este circuito definimos las rotaciones condicionales como:

$$R_k \equiv \begin{pmatrix} 1 & 0 \\ 0 & \mathrm{e}^{2\pi i/2^k} \end{pmatrix}$$

La primera puerta de Hadamard actúa sobre el bit más significativo, generando el estado:

$$|\psi_1\rangle = (H \otimes \mathbb{1} \otimes \mathbb{1})|\psi_0\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + \mathrm{e}^{2\pi i 0.x_2}|1\rangle\right) \otimes |x_1 x_0\rangle$$

Las siguientes puertas de rotación controlada agregan las fases (π/2) y (π/4) a $|x_2\rangle$ si los bits correspondientes están activos. De este modo:

$$|\psi_2\rangle = (R_2 \otimes \mathbb{1} \otimes \mathbb{1})|\psi_1\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + \mathrm{e}^{2\pi i 0.x_2x_1}|1\rangle\right) \otimes |x_1 x_0\rangle$$

$$|\psi_3\rangle = (R_3 \otimes \mathbb{1} \otimes \mathbb{1})|\psi_2\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + \mathrm{e}^{2\pi i 0.x_2x_1x_0}|1\rangle\right) \otimes |x_1 x_0\rangle$$

De la misma forma se aplica Hadamard y su rotación controlada al segundo bit:

$$|\psi_4\rangle = (\mathbb{1} \otimes H \otimes \mathbb{1})|\psi_3\rangle = \frac{1}{\sqrt{2^2}} \left(|0\rangle + e^{2\pi i 0.x_2 x_1 x_0}|1\rangle\right) \otimes \left(|0\rangle + e^{2\pi i 0.x_1}|1\rangle\right) \otimes |x_0\rangle$$

$$|\psi_5\rangle = (\mathbb{1} \otimes R_2 \otimes \mathbb{1})|\psi_4\rangle = \frac{1}{\sqrt{2^2}} \left(|0\rangle + e^{2\pi i 0.x_2 x_1 x_0}|1\rangle\right) \otimes \left(|0\rangle + e^{2\pi i 0.x_1 x_0}|1\rangle\right) \otimes |x_0\rangle$$

y por último la puerta de Hadamard al último bit:

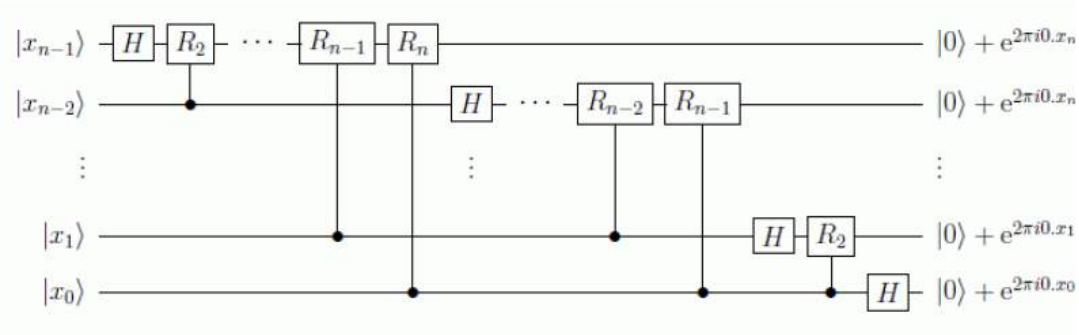
$$|\psi_6\rangle = (\mathbb{1} \otimes \mathbb{1} \otimes H)|\psi_5\rangle = \frac{1}{\sqrt{2^3}} \left(|0\rangle + e^{2\pi i 0.x_2 x_1 x_0}|1\rangle\right) \otimes \left(|0\rangle + e^{2\pi i 0.x_1 x_0}|1\rangle\right) \otimes \left(|0\rangle + e^{2\pi i 0.x_0}|1\rangle\right)$$

Vemos que para *n=3* se han necesitado 3 puertas de Hadamard y 3 puertas de rotación condicionada. Para el caso general se necesitan:

$$\text{Puertas QFT} = n \text{ Hadamard } + \binom{n}{2} \text{ Rotaciones} = n + \frac{n(n-1)}{2} = \frac{n(n+1)}{2} \text{ puertas}$$

Por tanto, el orden de cálculo de esta transformación cuántica es $O(n^2)=O((logN)^2)$, lo que denota una ganancia exponencial frente al mejor algoritmo clásico, que como comentamos era del orden $O(n^{2n})$. Por claridad no hemos incluido las puertas de intercambio que se necesitan (del orden de *n/2*) para obtener el orden correcto al final para nuestra transformada de Fourier tal como la definimos al principio.

Un posible circuito general para *n* qubits sería de la forma:



Una vez descrita la Transformada Cuántica de Fourier, ya estamos en condiciones de abordar el algoritmo de Shor, que es un algoritmo cuántico para descomponer en factores un número *N* en tiempo $O((\log N)^3)$ y espacio $O(\log N)$.

Como todos los algoritmos de computación cuántica, el algoritmo de Shor es probabilístico: da la respuesta correcta con alta probabilidad, y la probabilidad de fallo puede ser disminuida repitiendo el algoritmo.

El problema que intenta solucionar el algoritmo de Shor es que, dado un número entero *N*, intentamos encontrar otro número entero *p* entre *1* y *N* que divida *N*.

El algoritmo de Shor tiene dos partes:

- Una reducción del problema que implica una descomposición en factores para facilitar encontrar el orden, Esta primera parte se puede hacer en una computadora clásica.

- Un algoritmo cuántico para solucionar el problema de encontrar el orden.

La parte clásica del algoritmo de Shor puede describirse mediante el siguiente seudocódigo:

1. Escoger un número seudo-aleatorio $a < N$

2. Calcular el máximo común divisor: mcd (a,N). Esto se puede hacer usando el algoritmo de Euclides.

3. Si mcd (a, N) ≠ 1

    1. Entonces hemos encontrado un factor no trivial de $N \rightarrow$ Terminar.

    2. De lo contrario tendremos que encontrar el periodo de la función siguiente:

$$f(x) = a^x \bmod N$$

es decir el número entero más pequeño r para el cual

$$f(x + r) = f(x).$$

4. Si r es impar ir de nuevo al paso 1.

5. Si $a^{r/2} \equiv -1 \pmod{N}$ ir de nuevo al paso 1.

6. Los factores de N son el mcd ($a^{r/2} \pm 1$, N) $\rightarrow$Terminar.

La parte cuántica del algoritmo de Shor es la subrutina que nos permite encontrar el periodo y responde al seudocódigo siguiente:

1. Comenzar con un par de registros qubits de entrada y salida con $\log_2 N$ qubits cada uno, con $0 \leq x \leq N-1$, en el estado inicial siguiente:

$$N^{-1/2} \sum_x |x\rangle \, |0\rangle$$

2. Construir $f(x)$ como función cuántica y aplicarla al estado anterior para obtener:

$$N^{-1/2} \sum_x |x\rangle \, |f(x)\rangle$$

3. Aplicar la Transformada Cuántica de Fourier al registro de entrada.

$$U_{QFT}\,|x\rangle = N^{-1/2}\sum_{y} e^{2\pi i x y/N}\,|y\rangle$$

4. Obtenemos el estado siguiente:

$$N^{-1}\sum_{x}\sum_{y} e^{2\pi i x y/N}\,|y\rangle\,|f(x)\rangle$$

5. Realizar una medición. Obtenemos un cierto resultado *y* en el registro de entrada y $f(x_0)$ en el registro de salida. Puesto que *f* es periódica, la probabilidad de medir cierto *y* viene dada por

$$N^{-1}\left|\sum_{x:\, f(x)=f(x_0)} e^{2\pi i x y/N}\right|^2 = N^{-1}\left|\sum_{b} e^{2\pi i (x_0+rb) y/N}\right|^2$$

El análisis muestra ahora que cuanto más alta es esta probabilidad, tanto más el yr/N es cercano a un número entero.

6. Convertir y/N en una fracción irreducible y extraer el denominador r', que es un candidato a *r*.

7. Si $f(x) = f(x + r') \rightarrow$ Terminar.

8. Si $f(x) \neq f(x + r') \rightarrow$ Obtener más candidatos a *r* usando valores cercanos a *y*, o múltiplos de r'.

9. Si cualquier candidato cumple las condiciones → Terminar.

10. Si ningún candidato cumple las condiciones → Volver de nuevo al paso 1 del subprograma.

El algoritmo de Shor no es trivial. Trataremos de explicarlo. La primera parte del algoritmo convierte el problema de descomponer en factores en el

problema de encontrar el período de una función, y se puede implementar clásicamente. La segunda parte encuentra el período usando la transformada de Fourier cuántica.

Para la obtención de factores a partir del período consideramos que los números enteros menores que *N* y co-primos con *N* forman un grupo finito bajo multiplicación módulo *N*, que se denota típicamente $(\mathrm{Z}/N\,\mathrm{Z})^{\times}$. Al final del paso 3 tenemos un número entero *a* en este grupo. Puesto que el grupo es finito, *a* debe tener un orden finito *r*, el número entero positivo más pequeño tal que

$$a^r \equiv 1 \bmod N.$$

Por lo tanto, $N\,|(a^r - 1)$. Supongamos que podemos obtener *r*, y es par. Entonces:

$$a^r - 1 = (a^{r/2} - 1)(a^{r/2} + 1) \equiv 0 \bmod N$$

$$\Rightarrow N\,|(a^{r/2} - 1)(a^{r/2} + 1).$$

r es el número entero positivo *más pequeño* tal que $a^r \equiv 1$, así que *N* no puede dividir a $(a^{r/2} - 1)$. Si *N* tampoco divide $(a^{r/2} + 1)$, entonces *N* debe tener un factor común no trivial con $(a^{r/2} - 1)$ y $(a^{r/2} + 1)$.

Comprobamos lo anterior del siguiente modo:

Sea u = $(a^{r/2} - 1)$ y sea v =$(a^{r/2} + 1)$. $N \mid uv$, luego $kN = uv$ para un cierto número entero *k*. Supongamos que el mcd $(u, N) = 1$; entonces $mu + nN = 1$ para ciertos números enteros *m* y *n* (ésta es una propiedad del máximo común divisor.) Multiplicando ambos lados por *v*, encontramos que $mkN + nvN = v$, luego $N\,|v$. Por contradicción, mcd $(u, N) \neq 1$. Utilizando un argumento similar concluimos que mcd $(v, N) \neq 1$.

De este modo obtenemos una factorización de *N*. Si *N* es el producto de dos primos, esta es la *única* factorización posible.

Por otra parte, para encontrar el período, el algoritmo de Shor utiliza la capacidad de una computadora cuántica de estar en muchos estados simultáneamente. Los físicos llaman a este comportamiento superposición cuántica. Para computar el período de una función *f*, evaluamos la función en todos los puntos simultáneamente. Sin embargo, la física cuántica no permite que tengamos acceso a toda esta información directamente. Una medición cuántica dará solamente uno de todos los valores posibles destruyendo los demás. Por lo tanto tenemos que transformar cuidadosamente la superposición a otro estado que devuelva la respuesta correcta con alta probabilidad. Para ello se emplea la Transformada Cuántica de Fourier.

Shor tuvo que solucionar así tres problemas de implementación. Todos tuvieron que ser implementados "en una versión lo más rápida posible", lo que significa ejecutarlos con un número de puertas cuánticas que sea polinómico en log*N*. Esto implica:

1. Crear una superposición de estados, que puede hacerse aplicando las puertas de Hadamard a todos los qubits en el registro de entrada. Otro enfoque sería utilizar la transformada de Fourier cuántica.

2. Implementar la función *f* como una transformada cuántica. Para ello, Shor utilizó exponenciación por cuadrados para su transformación modular de la exponenciación.

3. Realizar una transformada de Fourier cuántica. Para ello, usando puertas controladas NOT y puertas de una sola rotación de qubit Shor diseñó un circuito para la transformada de Fourier cuántica que usa exactamente $((\log N)^2)$ puertas.

Después de todas estas transformaciones una medición dará una aproximación al período *r*. Por simplicidad asumiremos que hay una *y* tal que *yr/N* es un número entero. Entonces la probabilidad de medir *y* es 1. Para ver esto notemos que:

$$e^{2\pi ibyr/N} = 1$$

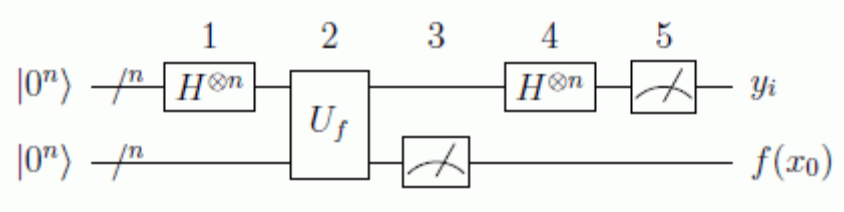
para todos los números enteros *b*. Por lo tanto la suma que nos da la probabilidad de la medición *y* será *N/r* puesto que *b* toma aproximadamente *N/r* valores y así la probabilidad es *1/r*. Hay *r*, *y* tales que *yr/N* es un número entero, luego la suma de las probabilidades es 1.

Algoritmo de Simon

Este algoritmo fue propuesto por Daniel Simon en 1994, y trata de encontrar el periodo de una función vectorial booleana del tipo:

$$f : \{0,1\}^n \longrightarrow \{0,1\}^n$$

Se puede probar que, de nuevo clásicamente, habría que evaluar *f* sobre la mitad más uno de los elementos del dominio ($2^{n-1}+1$), es decir, el coste sería exponencial. Incluso con un algoritmo probabilista no podríamos ir más allá de las $2^{n/2}$ consultas. Veremos que en este caso la ganancia cuántica es clara ya que bastará evaluar U_f unas cuantas veces (del orden de *n)*, para encontrar el periodo con una buena cota de aproximación. Analizamos el siguiente circuito:



Los dos registros del estado inicial tienen *n* qubits:

$$|\psi_0\rangle = |0^n\rangle_s \otimes |0^n\rangle_t$$

Aplicando los operadores de Hadamard al primer registro obtenemos:

$$|\psi_1\rangle = \left(H^{\otimes n} \otimes \mathbb{1}^{\otimes n}\right)|\psi_0\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle \otimes |0^n\rangle$$

Si hacemos actuar ahora la función obtenemos, dado que todos los bits del registro están a cero:

$$|\psi_2\rangle = U_f|\psi_1\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle \otimes |f(x)^n\rangle$$

Ahora añadimos un paso que no es necesario pero sí es muy útil desde el punto de vista pedagógico. Se trata de medir el segundo registro y obtener uno de los posibles valores de la función, digamos $f(x_0)$, quedando el estado colapsado a:

$$|\psi_3\rangle = M_2|\psi_2\rangle = \frac{1}{\sqrt{2}}(|x_0\rangle + |x_0 \oplus T\rangle) \otimes |f(x_0)\rangle$$

en donde por supuesto suponemos el estado más general combinación de x_0 y x_0+T, dado que sólo conocemos el valor de la función. Si ahora aplicamos Hadamard sobre los primeros qubits se obtiene:

$$|\psi_4\rangle = \left(H^{\otimes n} \otimes \mathbb{1}^{\otimes n}\right)|\psi_3\rangle = \frac{1}{\sqrt{2^{n+1}}} \sum_{y=0}^{2^n-1} \left[(-1)^{x_0 \cdot y} + (-1)^{(x_0 \oplus T)\cdot y}\right] |y\rangle \otimes |f(x_0)\rangle$$

en donde lógicamente ahí sólo sobrevivirán los términos que cumplen que $T \cdot y = 0$, ya que el resto interferirá destructivamente. Si ahora medimos el primer registro:

$$|\psi_5\rangle = M_1|\psi_4\rangle = |y_i\rangle \otimes |f(x_0)\rangle$$

obtendremos para cada estado y_i una probabilidad:

$$Prob(y_i/T \cdot y_i = 0) = \left| \frac{1}{\sqrt{2^{n+1}}} (-1)^{x_0 \cdot y_i} [1 + (-1)^{T \cdot y_i}] \right|^2 =$$

$$= \frac{1}{2^{n+1}} \left| 1 + (-1)^{T \cdot y_i} \right|^2 = \begin{cases} \frac{1}{2^{n-1}} & si \quad T \cdot y_i = 0 \\ 0 \quad si \quad T \cdot y_i \neq 0 \end{cases}$$

Repitiendo estos pasos del orden de *n* veces obtenemos *n* vectores y_i que nos darán un sistema lineal homogéneo de ecuaciones cuya solución no trivial nos dará las componentes de *T*:

$$T \cdot y_i = 0 \quad i = 1, \ldots, n$$

que podemos resolver con un algoritmo clásico obteniendo *T* con un orden *O(n)* de repeticiones. Como se ha dicho, este algoritmo es exponencialmente más eficiente que cualquier algoritmo clásico, incluso de tipo aleatorio.

Algoritmo de Grover

El algoritmo de Grover es un algoritmo cuántico para la búsqueda en una secuencia no ordenada de datos con *N* componentes en un tiempo $O(N^{1/2})$, y con una necesidad adicional de espacio de almacenamiento de $O(\log N)$ y fue propuesto por Lov K. Grover en 1996.

En una búsqueda normal de un dato, si tenemos una secuencia desordenada se debe realizar una inspección lineal que necesita un tiempo de *O (N)*, por lo que el algoritmo de Grover supone una mejora bastante sustancial que evita, además, la necesidad de la ordenación previa. No obstante, la ganancia obtenida es "sólo" de la raíz cuadrada, lo que contrasta con otras mejoras de los algoritmos cuánticos que obtienen mejoras de orden exponencial sobre sus contrapartidas clásicas.

Al igual que otros algoritmos de naturaleza cuántica, el algoritmo de Grover es un algoritmo de carácter probabilístico, por lo que produce la respuesta correcta con una determinada probabilidad de error que, no obstante, puede obtenerse tan baja como se desee por medio de iteraciones.

Aunque el propósito del algoritmo es, como ha sido indicado, la búsqueda en una secuencia, se podría describir de una manera más adecuada como la "inversión de una función". Así, si tenemos la función *y=f (x)*, que puede ser evaluada en un computador cuántico, este algoritmo nos permite calcular el valor de *x* cuando se nos da como entrada el valor de *y*. Además, invertir una función puede relacionarse con la búsqueda en una secuencia si consideramos que la misma es una función que produce el valor de *y* como la posición ocupada por el valor *x* en dicha secuencia.

El algoritmo de Grover también se puede utilizar para el cálculo de la media y la mediana de un conjunto de números, y para resolver otros problemas de naturaleza análoga. También se puede utilizar para resolver algunos problemas de naturaleza NP-completa por medio de inspecciones exhaustivas en un espacio de posibles soluciones. Esto resulta en una apreciable mejora sobre soluciones clásicas. El algoritmo de Grover tiene las siguientes fases:

Inicialización: Se considera una secuencia desordenada con N componentes. El algoritmo requiere un espacio de estados *N*-dimensional H, que puede ser modelado con $\log_2 N$ qubits. Numeremos las entradas de la secuencia con 0, 1,... (*N*-1); y seleccionemos un observable *Ω*, actuando sobre H, con *N* autovalores distintos conocidos. Cada uno de los autovalores de *Ω* codifica una d**e** las entradas de la secuencia de una forma que se describirá más adelante. Denotaremos los autoestados en la forma:

$$\{|0\rangle, |1\rangle, \cdots, |N-1\rangle\}$$

y los autovalores correspondientes como:

$$\{\lambda_0, \lambda_1, \cdots, \lambda_{N-1}\}$$

Ahora tomamos un operador unario, U_ω, que actúa como una subrutina que compara las diferentes entradas de acuerdo al criterio de búsqueda. El algoritmo no especifica cómo funciona la subrutina, pero debe ser una subrutina *cuántica* que trabaje bajo una superposición de estados. Además, debe actuar de manera especial sobre uno de los autoestados, $|\omega\rangle$, que corresponderá con la entrada que satisface el criterio de búsqueda. Más concretamente, requeriremos U_ω con los siguientes efectos:

$$U_\omega|\omega\rangle = -|\omega\rangle$$

$$U_\omega|x\rangle = |x\rangle \qquad \text{para todo } x \neq \omega$$

Además:

$$\langle\omega|\omega\rangle = 1.$$

$$\langle\omega|x\rangle = \langle x|\omega\rangle = 0.$$

$$U_\omega|\omega\rangle = (I - 2|\omega\rangle\langle\omega|)|\omega\rangle = |\omega\rangle - 2|\omega\rangle\langle\omega|\omega\rangle = -|\omega\rangle.$$

$$U_\omega|x\rangle = (I - 2|\omega\rangle\langle\omega|)|x\rangle = |x\rangle - 2|\omega\rangle\langle\omega|x\rangle = |x\rangle.$$

Nuestro objetivo es identificar el autoestado, $|\omega\rangle$ o de manera equivalente, el autovalor ω, tal que U_ω actúa especialmente sobre él.

Iteraciones del algoritmo: Los pasos del algoritmo de Grover son los siguientes:

1. Inicializar el sistema al estado

$$|s\rangle = \frac{1}{\sqrt{N}} \sum_x |x\rangle$$

2. Realizar la siguiente iteración *r (N)* veces. Donde la función *r (N)* se describe más adelante.
    1. Aplicar el operador U_ω
    2. Aplicar el operador $U_s = 2\,|s\rangle\,\langle s| - I$.
3. Realizar la medida Ω. La medida corresponderá al valor λ_ω con una cierta probabilidad que se puede aproximar a 1, para un cierto N>>1. A partir de λ_ω, se puede obtener ω.

Podemos escribir las operaciones realizadas:

$$\langle s|s\rangle = 1$$

$$\langle \omega|s\rangle = \langle s|\omega\rangle = \frac{1}{\sqrt{N}}$$

$$U_\omega|s\rangle = (I - 2|\omega\rangle\langle\omega|)|s\rangle = |s\rangle - 2|\omega\rangle\langle\omega|s\rangle = |s\rangle - \frac{2}{\sqrt{N}}|\omega\rangle$$

$$U_s(|s\rangle - \frac{2}{\sqrt{N}}|\omega\rangle) = (2\,|s\rangle\,\langle s| - I)(|s\rangle - \frac{2}{\sqrt{N}}|\omega\rangle)$$

$$= 2\,|s\rangle\,\langle s|\,s\rangle - |s\rangle - \frac{4}{\sqrt{N}}\,|s\rangle\,\langle s|\,\omega\rangle + \frac{2}{\sqrt{N}}|\omega\rangle$$

$$= 2|s\rangle - |s\rangle - \frac{4}{\sqrt{N}} \cdot \frac{1}{\sqrt{N}}|s\rangle + \frac{2}{\sqrt{N}}|\omega\rangle = \frac{N-4}{N}|s\rangle + \frac{2}{\sqrt{N}}|\omega\rangle$$

$$= \frac{1}{N\sqrt{N}}\left((N-4)\sum_{x\neq w}|x\rangle + (3N-4)|\omega\rangle\right)$$

Después de aplicar los dos operadores U_ωy U_s , la amplitud del elemento buscado se ve incrementada. En esto consiste una iteración de Grover.

Número de iteraciones: Ahora consideramos el plano definido por |s⟩ y |ω⟩. Sea |ω^{x}⟩ que es perpendicular a |ω⟩. Entonces |ω⟩ es uno de los vectores base, y tenemos:

$$\langle \omega | s \rangle = \frac{1}{\sqrt{N}}$$

En términos geométricos, hay un ángulo (π/2 - θ) entre |ω⟩ y |s⟩, donde θ viene dado por:

$$\cos\left(\frac{\pi}{2} - \theta\right) = \frac{1}{\sqrt{N}}$$

$$\sin\theta = \frac{1}{\sqrt{N}}$$

El operador U_ω es un reflejo del hiperplano ortogonal a |ω⟩ para los vectores en el plano definido por |s⟩ y |ω⟩, además, actúa como un reflejo de la línea |ω^{x}⟩. El operador U_s es un reflejo de la línea |s⟩.

Entonces, el vector de estado permanece en el plano de |s⟩ y |ω⟩ tras cada aplicación de U_s y tras cada aplicación de U_ω, y se puede comprobar que el operador $U_s U_\omega$ de cada paso de iteración rota el vector de estado en un ángulo de 2θ hacia |ω⟩.

El algoritmo se detendrá cuando el vector de estado se acerca a |ω⟩ tras lo cual las siguientes iteraciones rotan el vector de estado *fuera* de |ω⟩, reduciendo la probabilidad de obtener la respuesta correcta. El número de iteraciones necesarias es dado por *r*. Para alinear correctamente el vector de estado con |ω⟩, necesitamos:

$$\frac{\pi}{2} - \theta = 2\theta r$$

$$r = \frac{(\frac{\pi}{\theta} - 2)}{4}$$

Una consideración es que *r* debe ser entero, por lo que, en general, *r* será el entero más cercano a $(\pi/\theta - 2)/4$. Entonces, el αngulo entre $|\omega\rangle$ y el vector de estado final es $O(\theta)$, y la probabilidad de obtener una respuesta incorrecta es:

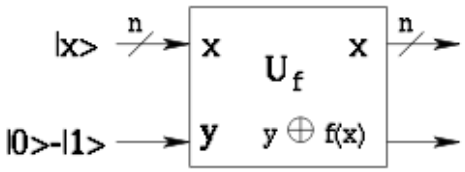
$$O(1 - \cos^2\theta) = O(\sin^2\theta).$$

Entonces, para $N>>1$, $\theta \approx N^{-1/2}$ tenemos

$$r \to \frac{\pi\sqrt{N}}{4}$$

Además, la probabilidad de obtener una respuesta incorrecta será $O(1/N)$, que tiende a 0 para un valor de N suficientemente elevado.

Implementación: Supongamos que tenemos una secuencia de 2^n elementos que vamos a referenciar por su índice *x*. Supongamos también que disponemos de una función *f (x)* que nos dice si el valor almacenado en la posición *x* es el que estamos buscando. En concreto sea *f (x)=1* para el valor buscado y *f (x)=0* para el resto. Consideremos el circuito cuántico que se muestra a acontinuación:



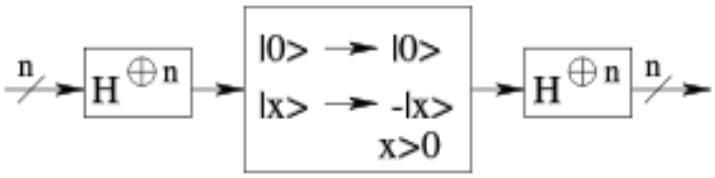
El funcionamiento de este bloque es el mismo que el correspondiente del algoritmo de Deutsch-Jozsa, y opera del siguiente modo:

$$U_f(|x\rangle(|0\rangle - |1\rangle)) = (-1)^{f(x)}|x\rangle|(|0\rangle - |1\rangle)$$

Puesto que el último estado no se modifica podemos ignorarlo y escribir:

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

Ahora procede realizar una inversión sobre la media, para lo cual empleamos el circuito de la figura:



Esta operación puede escribirse:

$$H^{\oplus n}(2|0\rangle\langle 0| - I)H^{\oplus n} = 2|\psi\rangle\langle\psi| - I$$

con

$$\psi = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle.$$

Claramente, esta operación se interpreta como inversión sobre la media, pues si la aplicamos sobre un estado genérico

$$a = \sum_{x=0}^{2^n-1} a_x|x\rangle$$

obtenemos:

$$(2|\psi\rangle\langle\psi| - I)(\sum_{x=0}^{2^n-1} a_x|x\rangle) = \sum_{x=0}^{2^n-1} (2\langle a\rangle - a_x)\,|x\rangle$$

,

en donde

$$\langle a\rangle = (1/2^n)\sum_{x=0}^{2^n-1} a_x$$

A continuación, la iteración de Grover según el circuito de la figura puede escribirse como:

$$G_f = (2|\psi\rangle\langle\psi| - I)U_f$$

Analizamos la primera iteración de Grover para lo cual preparamos un estado haciendo pasar el qubit $|0\rangle$ (realmente compuesto de *n* ceros) a través de una puerta de Hadamard:

$$|a\rangle = H^{\oplus n}|0\rangle = \frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}|x\rangle$$

Obtenemos:

$$|b\rangle = U_f(|a\rangle) = \frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}(-1)^{f(x)}|x\rangle$$

A continuación, aplicamos la inversión sobre la media y obtenemos:

$$|c\rangle = (2|\psi\rangle\langle\psi| - I)(|b\rangle) = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} (2 < b > -b_x)|x\rangle$$

$$= \frac{1}{2^n\sqrt{2^n}} \sum_{x=0}^{2^n-1} \left(2 \left(\sum_{y=0}^{2^n-1} b_y \right) - 2^n b_x \right) |x\rangle$$

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

Interpretemos ahora este resultado. Supongamos que en la posición x_i se encuentra el valor buscado, esto es, $f(x_i)=1$ y para el resto, $f(x)=0$, obtenemos:

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

$$= \frac{2^{n+1} + 2^n - 4}{2^n\sqrt{2^n}} |x_i\rangle + \frac{2^{n+1} - 2^n - 4}{2^n\sqrt{2^n}} \sum_{x=0, x\neq x_i}^{2^n-1} |x\rangle$$

Como puede observarse, el término que nos interesa aumenta su amplitud en comparación con los demás términos. Repitiendo esta operación en varias iteraciones este efecto de amplificación se verá incrementado. Si al final del algoritmo hacemos un medición, muy probablemente obtendremos el valor buscado.

Otra versión, quizás algo más gráfica del algoritmo de Grover es la que sigue a continuación, en donde también estudiamos el problema de una búsqueda en una base de datos, por ejemplo, la búsqueda de un teléfono en una guía telefónica sin conocer el nombre.

Si no sabemos nada sobre la estructura del espacio de soluciones estamos ante un problema desestructurado. Clásicamente el mejor algoritmo aleatorio nos llevaría a un coste de O(N) (si se quiere N/2 consultas de media) para una base de

datos de tamaño N. Con el algoritmo de Grover se mejora este resultado con una ganancia cuadrática de orden O(√N). El algoritmo de Grover es probabilístico, es decir, sólo da la respuesta correcta con cierta probabilidad (al contrario que el de Deutsch por ejemplo que era determinista).

Lógicamente nuestra proposición aquí está implementada por un oráculo que actúe de la siguiente forma:

$$f_{x_0}(x) : \{0,1\}^n \longrightarrow \{0,1\} / \begin{cases} 1 & si \quad x = x_0 \\ 0 & si \quad x \neq x_0 \end{cases}$$
$$U_{f_{x_0}} |x\rangle \otimes |y\rangle = |x\rangle \otimes |y \oplus f_{x_0}(x)\rangle$$

Por tanto asumimos que de entrada necesitaremos un registro fuente de -n- qubits tal que $N=2^n$ y uno adicional para almacenar la información de la función. (Nótese el hecho de que conocer de antemano x_0 no es lo mismo que reconocerlo entre un conjunto de estados).

La estrategia será preparar un estado superposición de todas las entradas del catálogo:

$$|\psi_1\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle$$

y después aplicar el oráculo que nos suministre todos los valores de la veracidad de la proposición sobre las entradas:

$$|\psi_2\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle \otimes |f(x)\rangle$$

Por último, tendremos que desbalancear los pesos estadísticos de forma que haya suficiente probabilidad de encontrar el estado $|x_0\rangle|1\rangle$. Para ello hay que utilizar las operaciones de cambio de signo e inversión sobre el promedio.

Se trata de aplicar un cambio de signo al elemento que cumpla la proposición, de forma que quede de algún modo marcado.

En el algoritmo clásico esto ya supondría, lógicamente, el haberlo encontrado, pero recordemos que aquí el elemento sigue dentro de un estado global que todavía no podremos medir por no tener la suficiente certeza de que el resultado nos va a dar nuestro elemento. El operador a implementar será de la forma:

$$U_{x_0} \equiv (\mathbb{1} - 2|x_0\rangle\langle x_0|)\,|x\rangle = \begin{cases} -|x_0\rangle & si \quad x = x_0 \\ |x_0\rangle & si \quad x \neq x_0 \end{cases}$$

La forma de implementar este algoritmo es preparando un estado destino, tal como se hace para el de Deutsch-Jozsa. De este modo el oráculo de la función nos dará un cambio de signo cuando ésta sea 1.

$$U_{f_{x_0}}|x\rangle \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right) = (-1)^{f_{x_0}(x)}|x\rangle \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right)$$

es decir, al no alterarse el segundo registro lo que se tiene es que:

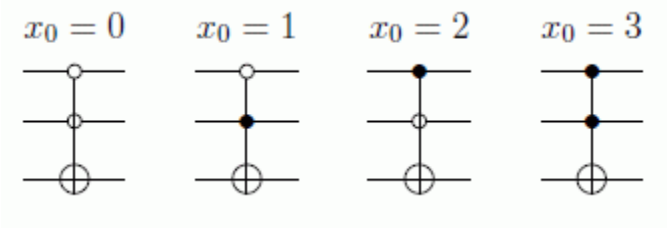
$$U_{f_{x_0}} = U_{x_0} \otimes \mathbb{1}$$

De esta forma el oráculo marca la solución del problema, mediante el operador cambio de signo:

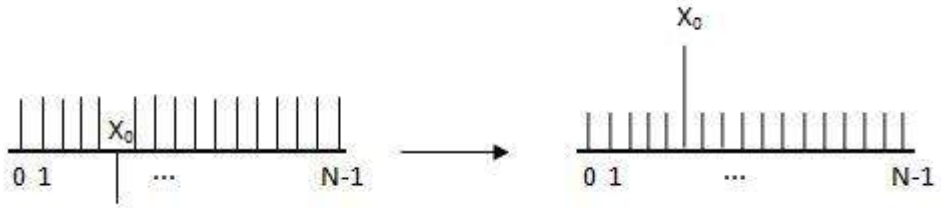
$$U_{x_0}|x\rangle = (-1)^{f_{x_0}(x)}|x\rangle = (\mathbb{1} - 2|x_0\rangle\langle x_0|)\,|x\rangle$$

$$U_{x_0} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots -1^{(x_0)} \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 1 \end{pmatrix}$$

Para N=4 por ejemplo este operador oráculo se puede implementar mediante las siguientes puertas de tipo Toffoli, dependiendo de si el x_0=0,1,2,3:



Ahora realizamos la inversión sobre el promedio que se ilustra en la figura:



Éste no es más que un algoritmo que superpone sobre la media la diferencia respecto de ésta. De este modo el valor negativo recientemente invertido aparecerá por encima de todos los demás. Para ello hay que usar el llamado por Grover operador de difusión:

$$D = H^{\otimes n}(-U_0)H^{\otimes n} = H^{\otimes n}(2|0\rangle\langle 0| - \mathbb{1})H^{\otimes n}$$

El efecto de este operador es equivalente a realizar una reflexión respecto del estado:

$$|\psi_1\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle$$

es decir

$$D = 2|\psi_1\rangle\langle\psi_1| - \mathbb{1} = \begin{pmatrix} \frac{2}{N} - 1 & \frac{2}{N} & \cdots & \frac{2}{N} \\ \frac{2}{N} & \frac{2}{N} - 1 & \cdots & \frac{2}{N} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{2}{N} & \frac{2}{N} & \cdots & \frac{2}{N} - 1 \end{pmatrix}$$

lo que se deduce simplemente aplicando la definición del operador Walsh-Hadamard respecto al estado fundamental y calculando después los elementos de matriz respecto de la base computacional.

Lo más interesante de este operador es estudiar lo que hace con las componentes de cualquier entrada que se le ponga, del tipo general:

$$|\psi\rangle = \sum_x \alpha_x |x\rangle$$

cuya acción es

$$\begin{aligned} D|\psi\rangle = (2|\psi_1\rangle\langle\psi_1| - \mathbb{1})|\psi\rangle &= \left(\frac{2}{N}\sum_x |x\rangle \sum_{x'} \langle x'| - \mathbb{1}\right) \sum_{x''} \alpha_{x''} \left|x''\right\rangle = \\ &= \left(\frac{2}{N}\sum_{xx'x''} \alpha_{x''} |x\rangle \left\langle x'\right| x''\right\rangle - \sum_x \alpha_x |x\rangle\right) = \left(\frac{2}{N}\sum_{x'} \alpha_{x'} \sum_x |x\rangle - \sum_x \alpha_x |x\rangle\right) = \\ &= \left(2 < \alpha > \sum_x |x\rangle - \sum_x \alpha_x |x\rangle\right) = \sum_x \left(2 < \alpha > -\alpha_x\right) |x\rangle \end{aligned}$$

es decir, se produce un cambio de las amplitudes en la forma

$$\alpha_x \longrightarrow 2 < \alpha > -\alpha_x = < \alpha > +(< \alpha >) - \alpha_x)$$

Vamos a ver el ejemplo de la actuación del cambio de signo y esta inversión para el caso N=4 (que puede ser implementado con dos qubits). En el registro entrante las amplitudes de todos los estados son las mismas:

$$\alpha_x = \frac{1}{\sqrt{4}} = \frac{1}{2}$$
$$< \alpha >_{\psi_1} = \frac{1}{4}\left[\frac{1}{2} + \frac{1}{2} + \frac{1}{2} + \frac{1}{2}\right] = \frac{1}{2}$$

Se aplica ahora un cambio de signo (sea x_0=2=(10)):

$$< \alpha >_{\psi} = \frac{1}{4}\left[\frac{1}{2} + \frac{1}{2} - \frac{1}{2} + \frac{1}{2}\right] = \frac{1}{4}$$

y por último aplicamos el operador de difusión, con lo que las amplitudes se transforman en:

$$(\alpha_x)_{x \neq x_0} = \frac{1}{2} \longrightarrow 2 \cdot \frac{1}{4} - \frac{1}{2} = 0$$
$$(\alpha_x)_{x = x_0} = -\frac{1}{2} \longrightarrow 2 \cdot \frac{1}{4} + \frac{1}{2} = 1!!$$

```
Número de iteraciones aconsejadas con error 1/N (0.25) =1
        Numero de elementos 4 (2 qubits)      x_o
Iteración   0    ->      0.50000     0.50000
Iteración   1    ->      0.00000     1.00000
Iteración   2    ->     -0.50000     0.50000
Iteración   3    ->     -0.50000    -0.50000
Iteración   4    ->      0.00000    -1.00000
Iteración   5    ->      0.50000    -0.50000
Iteración   6    ->      0.50000     0.50000
Iteración   7    ->      0.00000     1.00000
Iteración   8    ->     -0.50000     0.50000
Iteración   9    ->     -0.50000    -0.50000
Iteración  10    ->      0.00000    -1.00000
```

Lo que indica que en una sola iteración ya tenemos la certeza absoluta de encontrar el elemento, frente a una media de 2 en el caso clásico. En la figura anterior se ven las primeras iteraciones en este caso (y lo perjudicial que puede ser calcular de más). En general podemos decir que el aumento que se produce en la inversión temporal es aproximadamente:

$$\frac{1}{\sqrt{N}} \longrightarrow \frac{1}{\sqrt{N}} + \left(\sim \frac{1}{\sqrt{N}}\right)_{\text{por iteración}}$$

Este planteamiento es una forma heurística de ver que, en efecto, el método de Grover es de orden $O(\sqrt{N})$, dado que con ese orden de iteraciones llegaríamos a la unidad. En realidad esto no es así dado que las sucesivas iteraciones no tienen el mismo comportamiento, por eso va a ser tan importante calcular el número de iteraciones suficiente para una cierta probabilidad.

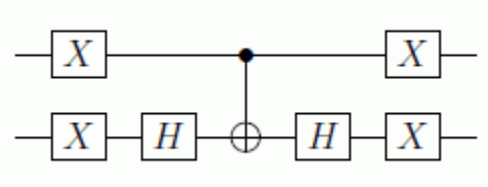
Para implementar este algoritmo basta con saber cómo se implementa la inversión respecto del eje $|0\rangle$ dada por $-U_0$. Para ello se usa el hecho de que:

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

A partir de esto podemos construir, con ayuda de puertas CNOT o Toffoli controladas, el operador que actúa reflejando el último vector de la base $|111\cdots1\rangle$. Si nos ayudamos también de puertas X podemos dar la vuelta al estado de forma que lo que gire sea el primero. Por ejemplo, para dos qubits tendríamos:

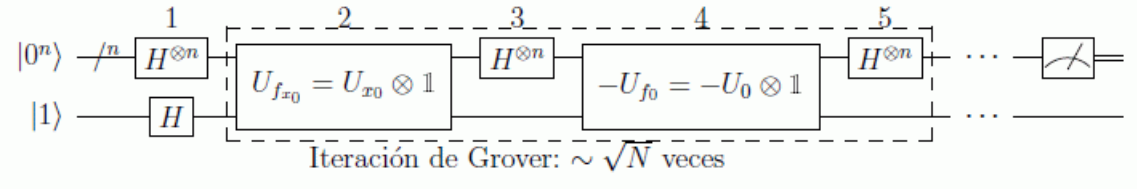
$$U_0 = (X \otimes X)(\mathbb{1} \otimes H)U_{CNOT}(\mathbb{1} \otimes H)(X \otimes X)$$

y el circuito correspondiente a U_0 sería:



El orden de cálculo de este algoritmo equivale al número de puertas ternarias de Toffoli necesarias para implementar una puerta de Toffoli de n-bits, es decir O(n)=O(log N).

A continuación describimos el algoritmo:



Partimos de un estado entrante de la forma:

$$|\Psi_0\rangle = |0^n\rangle \otimes |1\rangle$$

y al pasar por las puertas de Hadamard se tiene:

$$|\Psi_1\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle \otimes \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$$

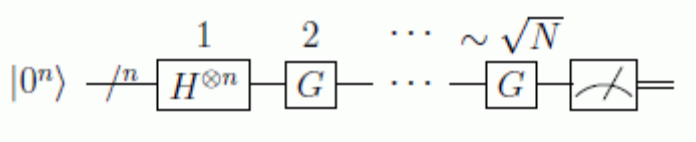
Como a partir de aquí el segundo registro no se va a modificar, es útil trabajar sólo con el primero, lo que implica empezar con el estado:

$$|\psi_1\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle$$

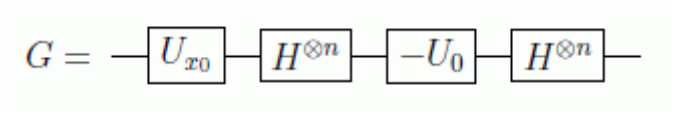
y estudiar su evolución bajo las iteraciones del operador de Grover:

$$\boxed{G = (2|\psi_1\rangle\langle\psi_1| - \mathbb{1})(1 - 2|x_0\rangle\langle x_0|) = DU_{x_0}}$$

De esta forma también podemos simplificar su representación gráfica:



donde



Por tanto la primera reflexión respecto a $|x_0\rangle$ nos dará en el primer registro el resultado:

$$U_{x_0}|\psi_1\rangle = (\mathbb{1} - 2|x_0\rangle\langle x_0|)|\psi_1\rangle = \frac{1}{\sqrt{N}} \left[\sum_{x \neq x_0} |x\rangle - |x_0\rangle \right]$$

y la inversión sobre el promedio nos da:

$$|\psi_2\rangle = DU_{x0}|\psi_1\rangle = (2|\psi_1\rangle\langle\psi_1| - \mathbb{1})\frac{1}{\sqrt{N}}\left[\sum_{x\neq x_0}|x\rangle - |x_0\rangle\right] =$$
$$= \frac{1}{\sqrt{N}}\left[\sum_{x\neq x_o}\left(1-\frac{4}{N}\right)|x\rangle + \left(3-\frac{4}{N}\right)|x_0\rangle\right]$$

en donde se ve claramente que para N=4 con una sola iteración ya se puede medir.

Si continuamos el siguiente paso nos daría:

$$|\psi_3\rangle = G|\psi_2\rangle = \frac{1}{\sqrt{N}}\left[\sum_{x\neq x_o}\left(1-\frac{12}{N}+\frac{16}{N^2}\right)|x\rangle + \left(5-\frac{20}{N}+\frac{16}{N^2}\right)|x_0\rangle\right]$$

La expresión general es difícil de compactar con fracciones polinómicas, pero tiene una expresión trigonométrica particularmente sencilla:

$$\frac{1}{\sqrt{N-1}}cos\left[(2k+1)arc\,cos\left(\frac{1}{\sqrt{N}}\right)\right]\sum_{x\neq x_0}|x\rangle + sen\left[(2k+1)arc\,sen\left(\frac{1}{\sqrt{N}}\right)\right]|x_0\rangle$$

Por otra parte, la interpretación geométrica es la siguiente: El estado $|\psi_1\rangle$ se puede separar mediante una base reducida en dos estados ortonormales:

$$|\psi_1\rangle = \sqrt{\frac{N-1}{N}}|x_\perp\rangle + \frac{1}{\sqrt{N}}|x_0\rangle \;/\; |x_\perp\rangle \equiv \frac{1}{\sqrt{N-1}}\sum_{x\neq x_0}|x\rangle$$

De forma que U_{x0} es una simetría respecto a $|x_\perp\rangle$ y D es una simetría respecto a $|\psi_1\rangle$ que en este caso tiene la forma:

$$D = 2\begin{pmatrix} \sqrt{\frac{N-1}{N}} \\ \frac{1}{\sqrt{N}} \end{pmatrix}\begin{pmatrix} \sqrt{\frac{N-1}{N}} & \frac{1}{\sqrt{N}} \end{pmatrix} - \mathbb{1} = \begin{pmatrix} 1-\frac{2}{N} & \frac{2\sqrt{N-1}}{N} \\ \frac{2\sqrt{N-1}}{N} & \frac{2}{N}-1 \end{pmatrix}$$

Por geometría básica se sabe que la composición de dos reflexiones de ejes secantes es un giro de ángulo doble al que forman los ejes. Sea ese ángulo de giro θ, luego el que forman los ejes será la mitad, y se cumple que:

$$cos\left(\frac{\theta}{2}\right) = \langle x_\perp |\, \psi_1 \rangle$$

luego

$$cos\left(\frac{\theta}{2}\right) = \sqrt{\frac{N-1}{N}} \quad y \quad sen\left(\frac{\theta}{2}\right) = \frac{1}{\sqrt{N}}$$

Si decimos que γ=θ/2 entonces el estado inicial es:

$$|\psi_1\rangle = cos(\gamma)|x_\perp\rangle + sen(\gamma)|x_0\rangle$$

y la aplicación del algoritmo supone un giro de θ=2γ en ese plano, es decir

$$|\psi_2\rangle = cos(3\gamma)|x_\perp\rangle + sen(3\gamma)|x_0\rangle$$

por lo que después de k interacciones tendremos

$$|\psi_{k+1}\rangle = cos((2k+1)\gamma)|x_\perp\rangle + sen((2k+1)\gamma)|x_0\rangle$$

Para calcular el número óptimo de iteraciones razonamos del siguiente modo: Es obvio que para rotar completamente el estado $|\psi_i\rangle$ a $|x_0\rangle$ se debe cumplir que...

$$sen[(2k_0+1)\gamma] = 1 \Rightarrow (2k_0+1)\gamma = \frac{\pi}{2} \Rightarrow k_0 = \frac{\pi}{4\gamma} - \frac{1}{2}$$

Expresión que no es entera, y por tanto habrá que aproximar, siendo en este paso cuando sale la dependencia del algoritmo en cuanto al orden O(N).

$$k_{aprox} = \left\lfloor \frac{\pi}{4\gamma} \right\rfloor \leq \left\lfloor \frac{\pi}{4sen(\gamma)} \right\rfloor = \left\lfloor \frac{\pi\sqrt{N}}{4} \right\rfloor$$

Vamos a calcular la probabilidad de fallo, definida como

$$Prob = cos^2[(2k_{aprox}+1)\gamma] = sen^2\left[\frac{\pi}{2} - (2k_{aprox}+1)\gamma\right]$$

para lo cual nos basamos en que:

$$|k_0 - k_{aprox}| \leq \frac{1}{2} \Rightarrow \left|\frac{\pi}{2} - (2k_{aprox}+1)\gamma\right| = |(2k_0+1)\gamma - (2k_{aprox}+1)\gamma| = \\ = |2\gamma(k_0 - k_{aprox})| \leq \gamma$$

Por tanto:

$$Prob \leq sen^2\gamma = \frac{1}{N}$$

```
Número de iteraciones aconsejadas con error 1/N (0.015625) =6
        Numero de elementos 64 (6 qubits)      x_o
        Iteración   0    ->      0.12500     0.12500
        Iteración   1    ->      0.11720     0.36719
        Iteración   2    ->      0.10210     0.58643
        Iteración   3    ->      0.08054     0.76901
        Iteración   4    ->      0.05399     0.90354
        Iteración   5    ->      0.02406     0.98159
        Iteración   6    ->     -0.00736     0.99829
        Iteración   7    ->     -0.03833     0.95260
```

Esta es la razón de que la probabilidad de fallo después de $\pi\sqrt{N}/4$ iteraciones sea 1/N. Normalmente la probabilidad de encontrar el elemento es mayor que esta cota, como se aprecia en la figura anterior para 64 elementos, en donde aunque esta probabilidad es de 0.016, en realidad en la iteración recomendada tenemos una amplitud de 0.998, con lo que el margen de error probabilístico es de 0.004.

Para introducir la dependencia con el número de soluciones hay que definir los conjuntos:

$$X_0 = \{x/f(x) = 0\}$$
$$X_1 = \{x/f(x) = 1\}$$

de forma que, si -s- es el número de soluciones:

$$|x_\perp\rangle = \frac{1}{\sqrt{N-s}} \sum_{x \in X_0} |x\rangle$$
$$|x_0\rangle = \frac{1}{\sqrt{s}} \sum_{x \in X_1} |x\rangle$$

resulta que

$$|\psi_1\rangle = \sqrt{\frac{N-s}{N}}|x_\perp\rangle + \sqrt{\frac{s}{N}}|x_0\rangle$$

Y de nuevo tendremos la misma interpretación trigonométrica:

$$|\psi_{k+1}\rangle = cos[(2k+1)\gamma]|x_\perp\rangle + sen[(2k+1)\gamma]|x_0\rangle$$

pero ahora el ángulo se define como:

$$cos\left(\frac{\theta}{2}\right) = cos(\gamma) = \langle x_\perp |\, \psi_1 \rangle = \sqrt{\frac{N-s}{N}} \Rightarrow \gamma = arc\, sen\left(\sqrt{\frac{s}{N}}\right)$$

Así que por la misma razón ahora el orden de iteraciones va con O(√N/s):

$$\left\lfloor \frac{\pi}{4\gamma} \right\rfloor \leq \left\lfloor \frac{\pi}{4sen(\gamma)} \right\rfloor = \left\lfloor \frac{\pi}{4}\sqrt{\frac{N}{s}} \right\rfloor$$

Casos particulares interesantes del modelo, que merecen la pena ser comentados son los siguientes:

- s=1. Ya hemos visto que si N es grande:

$$\left\lfloor \frac{\pi}{4\gamma} \right\rfloor \simeq \frac{\pi\sqrt{N}}{4}$$

- s=N/4. En este caso sen(γ)=1/2 por lo que:

$$\left\lfloor \frac{\pi}{4\gamma} \right\rfloor = \left\lfloor \frac{3}{2} \right\rfloor = 1$$

lo que significa que con una sola iteración se consigue la solución.

- s=N/2. Entonces sen(γ)=1/√2 luego γ=π/4 y

$$\left\lfloor \frac{\pi}{4\gamma} \right\rfloor = 1$$

pero en este caso la aplicación del algoritmo no mejora la probabilidad de acierto clásica, que sigue siendo s/N=1/2:

$$|\psi_2\rangle = cos(3\gamma)|x_\perp\rangle + sen(3\gamma)|x_0\rangle = -\frac{1}{\sqrt{2}}|x_\perp\rangle + \frac{1}{\sqrt{2}}|x_0\rangle$$

- s>N/2. A partir de aquí, el número de iteraciones se va haciendo más grande y no se gana respecto al caso clásico, ya que el ángulo de giro cumple:

$$sen(\theta) = sen(2\gamma) = 2sen(\gamma)cos(\gamma) = \frac{2}{N}\sqrt{s(N-s)}$$

expresión que alcanza su valor máximo en s=N/2, con lo que después desciende y por tanto el número de iteraciones será mayor.

De todo esto se deduce que si a priori no se conoce el número de soluciones el algoritmo no es útil. En efecto, si por ejemplo $N=2^{20}$, ya hemos visto que una probabilidad de fallo menor que 2^{-20} nos la darían, para una solución, el siguiente número de iteraciones:

$$\left\lfloor \frac{\pi\sqrt{N}}{4} \right\rfloor = 804\ iteracciones$$

pero si mantenemos este número de iteraciones y resulta que existen s=4 soluciones, el error que se comete es muy grande, ya que en este caso deberíamos haber empleado 402 iteraciones, encontrando para el valor 804:

$$sen[1609 arc\, sen\left(\frac{1}{512}\right)] = -0{,}000987!$$

cuando teníamos que haber parado en el 402:

$$sen[805 arc\, sen\left(\frac{1}{512}\right)] = 1$$

Para hallar la relación de recurrencia del algoritmo de Grover, escribimos el estado en la forma:

$$|\psi_{k+1}\rangle = a_k \sum_{x \in X_0} |x\rangle + b_k \sum_{x \in X_1} |x\rangle$$

en donde la semilla es:

$$a_0 = b_0 = \frac{1}{\sqrt{N}}$$

Como en cada iteración las amplitudes cambian de signo y se produce la inversión en el promedio, las relaciones recursivas son:

$$a_{k+1} = 2M_k - a_k \qquad b_{k+1} = 2M_k + b_k$$
$$M_k = \frac{(N-s)a_k - sb_k}{N}$$

que podemos expresar en forma matricial del siguiente modo:

$$\begin{pmatrix} a_{k+1} \\ b_{k+1} \end{pmatrix} = \begin{pmatrix} \frac{N-2s}{N} & -\frac{2s}{N} \\ \frac{2N-2s}{N} & \frac{N-2s}{N} \end{pmatrix} \begin{pmatrix} a_k \\ b_k \end{pmatrix}$$

Por inducción se puede demostrar que la solución de este sistema es precisamente el resultado obtenido geométricamente:

$$a_k = \frac{1}{\sqrt{N-s}} cos((2k+1)\gamma) \qquad b_k = \frac{1}{\sqrt{s}} sen((2k+1)\gamma)$$

La generalización se puede plantear aprovechando el hecho de que, según la Transformada Cuántica de Fourier:

$$|\widetilde{x}\rangle \equiv U_{QFT}|0\rangle = \frac{1}{\sqrt{N}}\sum_x |x\rangle = |\psi_1\rangle \Rightarrow |\psi_1\rangle\langle\psi_1| = \left|\widetilde{0}\right\rangle\left\langle\widetilde{0}\right|$$

por tanto la inversión sobre la media toma la misma forma matricial en la base de momentos que el cambio de signo en la de coordenadas. Si definimos:

$$\bar{P} = |\psi_1\rangle\langle\psi_1| = \frac{1}{N}\begin{pmatrix} 1 & 1 & \cdots & 1 \\ \cdots & \cdots & \cdots & \cdots \\ 1 & 1 & \cdots & 1 \end{pmatrix}$$

en la base transformada, encontramos la expresión:

$$\widetilde{\bar{P}} = U_{QFT}^{-1}\bar{P}U_{QFT} = |0\rangle\langle 0| \equiv P_0$$

Esto nos puede llevar a descubrir qué pasaría si se utiliza un proyector sobre otro estado de "momento", pero la cuestión queda fuera de los objetivos de este texto.

--..--

A MODO DE CONCLUSIÓN

En los últimos años, los ordenadores clásicos han experimentado un gran aumento en la velocidad de procesamiento. La miniaturización del tamaño de sus componentes ha facilitado el incremento de la densidad de los circuitos electrónicos que los integran. En 1995, Gordon Moore vaticinó que el número de transistores de un microprocesador se multiplicaría por dos cada dos años. Y esta ley, que se ha venido cumpliendo hasta ahora, cuenta con una limitación: cuando el tamaño de los transistores presenta medidas atómicas las leyes más fundamentales de la física cambian. Los electrones experimentan comportamientos cuánticos y pueden moverse entre distintas líneas de corriente por "efecto túnel". Esto produce la aparición de fugas que interfieren en el funcionamiento del circuito.

El progreso técnico llega a su fin. Pero los principios de la cuántica, que limitan la dimensión de los microcircuitos de los ordenadores clásicos, son el germen de una nueva revolución computacional. La física cuántica deja de ser una teoría abstracta, misteriosa y anti-intuitiva para convertirse en útil, para ser clave en el desarrollo de una futura teoría de la información. Los primeros físicos teóricos que, en las décadas de 1970 y 1980, propusieron aplicar los fenómenos cuánticos al terreno de la computación fueron Richard Feynmann, Paul Benioff, David Deutsch y Charles Bennett. La resolución de problemas en un ordenador se realiza a través de algoritmos que son conjuntos precisos de instrucciones. Su eficiencia se evalúa a partir del ritmo en el que se incrementa el tiempo de resolución del problema, a medida que aumenta el tamaño de los datos de entrada.

La mecánica cuántica ha introducido nuevos algoritmos que permiten resolver problemas a velocidades increíblemente superiores a las de los más avanzados ordenadores actuales.

David Deutsch, junto a Richard Jozsa, fueron pioneros en el campo de la computación cuántica al formular en 1992 el primer algoritmo cuántico: el

algoritmo de Deutsch-Jozsa. Éste fue mejorado en 1998 por Richard Cleve, Artur Ekert, Chiara Macchiavello y Michele Mosca.

En 1994, Peter Shor, de los Laboratorios Bell, describió el primer gran algoritmo cuántico, diseñado para factorizar números grandes (de varios centenares de dígitos) en un tiempo record. La dificultad que tiene la computación clásica en la factorización de estos números se utiliza actualmente en los códigos de seguridad. Por tanto, el algoritmo de Shor, que puede realizar esta operación con extrema rapidez, se convierte en el sueño de cualquier hacker.

Dos años más tarde, en 1996, Lov Grover descubrió el segundo gran algoritmo cuántico que permite llevar a cabo búsquedas inversas en extensas bases de datos. El tiempo de ejecución es muy inferior al que se necesita en un ordenador convencional, si bien no se da una reducción de tiempo tan acusada entre los dos tipos de computación como en el caso del algoritmo de Shor. Por el momento, no se han descrito más algoritmos cuánticos. Esto se debe a que, si bien los ordenadores producen gran cantidad de operaciones simultáneas en paralelo, la medición obtiene un único resultado. Es decir, se pierde la información de las otras posibilidades asociadas al resto de estados superpuestos. Por tanto, cualquier nuevo algoritmo que se cree tendrá que expresar la información deseada en una sola medida.

En el cómputo cuántico la unidad mínima de información es el qubit (quantum bit) que, a diferencia del bit que sólo puede tomar los valores 0 y 1, se encuentra en una superposición simultánea de dos estados cuánticos**,** y en N qubits se encuentran simultáneamente superpuestos 2^N estados. Esta superposición cuántica permite la posibilidad de realizar un procesamiento paralelo a gran escala. Es decir, la capacidad operacional de un ordenador cuántico aumenta exponencialmente con el tamaño del mismo, el número de qubits.

El estado de un qubit puede verse como un punto en la superficie de una esfera (llamada esfera de Bloch). En esta representación los polos de la esfera

representan los bits clásicos “0″ y “1″ y todos los demás puntos son las distintas posibilidades que puede tomar un qubit.

Al margen de la superposición cuántica de estados, otro fenómeno clave que explica la gran potencia de los ordenadores cuánticos es el entrelazamiento. Dos sistemas cuánticos entrelazados mantienen un vínculo tal que, a pesar de la distancia que haya entre ellos, no pueden describirse separadamente. La aplicación estrella del entrelazamiento cuántico es la teletransportación. A partir de ésta, el cambio en el estado cuántico de uno de los sistemas se teletransporta instantáneamente al sistema cuántico lejano. Es importante señalar que lo que se teletransporta es la información, no la materia. Así pues, en el caso de la computación cuántica se transmiten qubits sin enviar qubits.

El principal problema al que tiene que hacer frente la computación cuántica es el efecto de la decoherencia. Ésta consiste en la pérdida de información del sistema debido a la interferencia del ambiente en la superposición de estados. En consecuencia, los modelos físicos deben cumplir unos requisitos imprescindibles para actuar como un ordenador cuántico. Por un lado, los qubits deben estar tan aislados del entorno como sea posible para evitar los efectos de la decoherencia y, por el otro, debe permitirse una interacción controlada con otros qubits para poder crear los estados entrelazados y, posteriormente, proceder a la lectura del resultado.

De lo expuesto, parece que los errores provocados por la decoherencia podrían ser nefastos para la consecución del cálculo; sin embargo, se están desarrollando métodos basados en la propia teoría cuántica para corregirlos. Recientemente se ha publicado un artículo en la revista Nature: “Quantum physics: Cruise control for a qubit,” por Howard M. Wiseman, sobre la implementación experimental de un método teórico de control realimentado de la decoherencia cúantica que había sido formulado en 2002.

Existen diversos sistemas físicos que cumplen con los requerimientos necesarios para ser ordenadores cuánticos. Se han hecho ordenadores de muy

pocos qubits y aún no se puede determinar con seguridad qué sistema físico es el idóneo.

Uno de los problemas que tienen algunos de los prototipos, es la falta de escalabilidad a ordenadores que cuenten con los qubits necesarios para desarrollar aplicaciones de interés práctico. Esto se debe a que, a medida que aumenta el número de qubits, se hace más difícil mantener su estabilidad y se requiere el desarrollo de complejos métodos de detección de errores.

En 1995 los físicos Juan Ignacio Cirac y Peter Zoller idearon el esquema básico para construir ordenadores cuánticos con trampas de iones. Explicaron cómo podía hacerse algo que hasta entonces era una idea abstracta, una mera entelequia teórica. Dieron un paso fundamental que, como veremos, acaba de recibir su justo reconocimiento.

Juan Ignacio Cirac Sasturain *(11 de octubre de 1965, Manresa, provincia de Barcelona, Cataluña) es un físico español reconocido por sus investigaciones en computación cuántica y óptica cuántica, enmarcadas en la teoría cuántica y en la física teórica. Desde 2001 es director de la División Teórica del Instituto Max-Planck de Óptica Cuántica (Max-Planck-Institut für Quantenoptik) en Garching, Alemania.*

En general, los principales candidatos a ordenadores cuánticos son los sistemas físicos óptico-cuánticos en los cuales los qubits son átomos (o iones) y su manipulación se realiza mediante luz láser; los sólidos cuyos qubits pueden ser pares de electrones en un lado u otro de un potencial, o bien, electrones en distintos estados de un punto cuántico y los sistemas basados en la resonancia magnética nuclear, en cuyo caso, los qubits son los átomos de una molécula y las lecturas de los resultados se obtienen mediante la técnica de resonancia magnética nuclear.

Los puntos cuánticos pueden imaginarse como “átomos artificiales” en los cuales los electrones confinados se encuentran en niveles energéticos similares a los que tendrían en un átomo pero en ausencia de campo electromagnético externo.

Lo cierto es que se sucede la publicación de artículos sobre implementaciones de qubits en sistemas físicos. Algunos de los más recientes han sido el almacenamiento de un qubit durante tres minutos en una memoria cuántica basada en silicio (artículo de la revista Science de Christoph Boehme, Dane R. McCamey, “Nuclear-Spin Quantum Memory Poised to Take the Lead,” del 8 Junio de 2012) y el entrelazamiento de un fotón al espín de un electrón confinado en un punto cuántico (artículo de la revista Nature de Sophia E. Economou, “Quantum physics: Putting a spin on photon entanglement,” del 15 de Noviembre de 2012).

Las investigaciones premiadas con el Nobel de Física de 2012 tienen una implicación directa en el avance de la computación cuántica, en la implementación física de los qubits. En palabras de la Academia Sueca: “Sus métodos innovadores han permitido dar los primeros pasos hacia la construcción de un nuevo tipo de ordenador súper-rápido basado en la física cuántica.”

Los galardonados son Serge Haroche, de la Escuela Normal Superior de París y David Wineland, del Instituto Nacional de Normas y Tecnología de EEUU en Maryland. Tal y como reza el comunicado de la Real Academia de Ciencias de Suecia: “Los premiados han abierto la vía a una nueva era de experimentación en la física cuántica al demostrar la observación directa de partículas cuánticas individuales sin destruirlas”. Ambos han logrado manipular sistemas cuánticos formados por una única partícula sin que ésta pierda sus propiedades cuánticas, que, por tanto, podrán medirse.

Wineland empleó fotones para medir el estado cuántico de átomos o iones atrapados en una trampa, mientras que Haroche lo consiguió mediante la

estrategia opuesta: creó una trampa para fotones y analizó sus propiedades cuánticas mediante átomos.

En realidad, las técnicas para atrapar las partículas cuánticas individuales (fotones o átomos/iones) no son la novedad, y ya habían sido premiadas, la aportación que ha valido el Nobel en esta ocasión es la posibilidad de medir y estudiar los estados cuánticos de las partículas atrapadas. Si recordamos la paradoja del gato de Schrödinger podemos decir que el gran logro ha sido saber si el gato estaba vivo o no sin abrir la caja.

La computación cuántica está de moda. El premio Wolf de 2013 ha sido concedido a Juan Ignacio Cirac y a Peter Zoller por sus "revolucionarias contribuciones teóricas al procesado de información cuántica, la óptica cuántica y la física de gases cuánticos." Estos premios son otorgados por la Fundación Wolf, que fue creada en 1975 por Ricardo Wolf, un inventor y diplomático de origen alemán, y se consideran la antesala de los Nobel. Los galardones se conceden en seis campos: agricultura, química, matemáticas, medicina, física y artes.

Como hemos visto, su contribución en la implementación física de los ordenadores cuánticos fue fundamental para el desarrollo de los mismos. Y su trabajo no se ha limitado al caso de la computación sino que han aplicado la teoría de la información cuántica a otros casos como es la construcción de simuladores.

En palabras de Cirac a la web de Physicsworld: "Es un gran honor recibir el premio Wolf. Creo que es justo decir que este premio también reconoce el trabajo de los científicos que han colaborado con nosotros. Solo somos dos representantes de los muchos científicos que han hecho grandes contribuciones al campo de la información cuántica, un campo en pleno auge, que sigue avanzando y atrayendo a muchas comunidades diferentes de científicos."

El profesor Juan Ignacio Cirac también ha sido galardonado con la medalla de honor del Instituo Niels Bohr "en reconocimiento a su verdaderamente

extraordinaria contribución al desarrollo de nuevas teorías sobre el futuro de las redes informáticas basadas en las leyes de la mecánica cuántica."

La medalla de honor se creó en 2010 con motivo de la celebración del 125 aniversario del nacimiento del físico Niels Bohr. Se otorga anualmente y premia a aquellos científicos que en sus investigaciones siguen el espíritu científico de Bohr: cooperación internacional e intercambio de conocimiento.

Como hemos comentado, por el momento los ordenadores cuánticos que se han construido disponen de un número muy limitado de qubits para resolver problemas de interés; sin embargo, la propia academia sueca ya apunta que "no hay motivo para pensar a priori que no sea posible conseguir estas operaciones con muchos más qubits".

Los continuos avances en las realizaciones prácticas de los diferentes modelos teóricos parecen indicar que el ordenador cuántico estará entre nosotros antes de lo que podíamos creer, pero aún así, es difícil aventurarse a dar una fecha aproximada. El propio Cirac, en una entrevista a ABC ha apuntado que es muy posible que aún se tarde varias décadas. En fin, en esta situación y recuperando las palabras del gran físico Niels Bohr: "Hacer predicciones es muy difícil, especialmente cuando se trata del futuro."

Niels Henrik David Bohr. *Nació en Copenhague. Tras doctorarse en la Universidad de Copenhague en 1911, e intentar la ampliación de estudios en el Cavendish Laboratory de Cambridge, el joven Bohr completó sus estudios en Mánchester, teniendo como maestro a Ernest Rutherford, con el que estableció una duradera relación científica y amistosa. En 1916, Bohr comenzó a ejercer como profesor de Física Teórica en la Universidad de Copenhague, consiguiendo los fondos para crear el Instituto Nórdico de Física Teórica, que dirigió desde 1920 hasta su fallecimiento en 1962.*

--..--

BIBLIOGRAFÍA

Una parte del material empleado en la elaboración de este texto ha sido obtenido a través de Internet (Palabras Clave: Computación Cuántica, Información Cuántica), y ensamblado posteriormente tras un laborioso proceso. Además, ha sido consultada la siguiente bibliografía específica:

- Barenco, A., Benett, C.H., Cleve, R., DiVincenzo, D.P., Margolus, N., Shor, P., Sleator, T., Smolin, J., Weinfurter, H., "Elementary Gates for Quantum Computation", Phys. Rev. A, vol. 52, pp 3457-3467, 1995.

- Benioff, P., "Quantum Mechanical Hamiltonian Models of Turing Machines", J. Stat. Phys., vol. 29, pp. 515-546, 1982.

- Bennett, C.H., "Logical Reversibility of Computation", IBM Journal of Research and Development, vol. 17, pp. 525-532, 1973.

- Bennett, C.H., "The Thermodynamics of Computation: A Review", Internat. J. Theoret. Phys., vol. 21, pp. 905-940, 1982.

- Bennett, C.H. and Landauer, R., "Fundamental Physical Limits of Computation", Scientific American, vol. 253, pp. 48-56, 1985.

- Bennett C.H. and Shor P. W., "Quantum Information Theory", IEEE Trans. Info. Theory, vol. 44, pp. 2724-2742, 1998.

- Bennett, C.H. and DiVincenzo, D.P., "Quantum Information and Computation", Nature, vol. 404, pp. 247-255, 2000.

- Desurvire, E., "Classical and Quantum Information Theory", Cambridge University Press. 2009.

- Deutsch, D., "Quantum Theory, the Church-Turing Principle and the Universal Quantum Computer", Proc. of the Royal Society of London, A400, pp. 97-117, 1985.

- Dieks, D., “Communication by EPR Devices”, Phys. Lett. A, vol. 92, pp. 271-272, 1982.
- Feynman, R.P., “Simulating Physics with Computers”, International Journal of Theoretical Physics, vol. 21, pp. 467-488, 1982.
- Feynman, R.P., “Conferencias Sobre Computación”, Crítica eds., 2003.
- Fredkin, E. and Toffoli, T., “Conservative Logic”, International Journal of Theoretical Physics, vol.21, pp. 219–253, 1982.
- Gisin, N., “Nonlocality Criteria for Quantum Teleportation”, Phys. Rev. Lett. A, vol. 210, pp. 151-156, 1996.
- Grupo de Computación Cuántica, Departamento de Matemática Aplicada, E.U. Informática, U. Politécnica Madrid, “Introducción al Modelo Cuántico de Computación”, TECHNICAL REPORT Nº 19, 2003.
- Landauer, R., “Irreversibility and Heat Generation in the omputing Process”, IBM Journal of Research and Development, vol. 5, pp. 183-191, 1961.
- Levine, I.N., “Química Cuántica”, AC eds., 1977.
- Meglicki, Z., “Quantum Computing without Magic: Devices”, The MIT Press, 2008.
- Nielsen, M.A. and Chuang, I.L., “Quantum Computation and Quantum Information”, Cambridge University Press, 2000.
- Toffoli, T., “Reversible Computing”, MIT Technical Report MIT/LCS/TM-151, 1980.
- Toffoli, T. and Margolus, N., “Cellular Automata Machines: A New Enviroment for Modelling”, The MIT Press, 1987.

- Wootters, W. K. and Zurek, W. H., “A Single Quantum Cannot be Cloned”, Nature, vol. 299, pp. 802, 1982.
- Yanofsky, N.S. and Mannucci, M.A., “Quantum Computing For Computer Scientists”, Cambridge University Press, 2008.