

Introducción al modelo cuántico de computación

Grupo de Computación Cuántica

TECHNICAL REPORT N° 19
Junio 2003

Grupo de Computación Cuántica

Dep. Matemática Aplicada. E.U. Informática
Ctra. de Valencia Km. 7, 28031 Madrid, Spain
e_mail: jglopez@eui.upm.es

Departamento de Matemática Aplicada
E.U. Informática. U. Politécnica Madrid
Ctra. de Valencia Km. 7
28031 Madrid, España

Introducción al modelo cuántico de computación*

Grupo de Computación Cuántica[†]

3 de junio de 2003

Abstract

En este artículo introducimos el modelo cuántico de computación que siguen la mayoría de los investigadores que trabajan en este tema. Su principal característica es la capacidad para realizar simultáneamente un número exponencial de operaciones. Esta propiedad, denominada paralelismo cuántico, permitió a P. W. Shor diseñar un algoritmo polinomial para factorizar números enteros. Este resultado es quizás el hito más notable de la computación cuántica. Sin embargo, es preciso mencionar que se trata de un modelo de computación teórico. Hasta ahora no se han construido ordenadores cuánticos que puedan aplicar este modelo de computación, aunque existe una intensa actividad investigadora en esta línea.

1 Introducción

La computación cuántica empezó a desarrollarse en la década de los ochenta a raíz de las propuestas de Paul Benioff, David Deutsch y Richard Feynman. En 1982 Benioff [3] y Feynman [7] sugirieron independientemente que, dado el elevado coste computacional del cálculo de la evolución de sistemas cuánticos, la evolución de estos sistemas se podría considerar como una herramienta de cálculo más que como un objeto a calcular. Poco después, en 1985, y también de forma independiente Deutsch [5] propone la búsqueda de un ordenador que sea capaz de simular eficientemente un sistema físico arbitrario. La conjunción de todas estas ideas han conducido a la concepción actual de ordenador cuántico.

Cuestionar el sistema de computación clásico, que cuenta con una sólida base teórica y con el aval de infinidad de aplicaciones en todos los ámbitos de la vida cotidiana, sólo tiene sentido si el modelo que se propone como alternativo es potencialmente mejor que el actual. Efectivamente así lo hacen Benioff, Deutsch y Feynman, fundamentando sus propuestas sobre la posibilidad de que los sistemas cuánticos tengan mayor potencia de cálculo que los clásicos. El argumento que todos utilizan para apuntar esta posibilidad es el hecho de que la simulación de un ordenador cuántico (sistema cuántico) en un ordenador clásico requiere una gran cantidad de operaciones.

El principal método para aumentar la capacidad de cálculo de un ordenador clásico es el procesamiento en paralelo. Los ordenadores que soportan este esquema de programación disponen de varios cientos o miles de procesadores. Sabemos que la capacidad de almacenamiento de información y la capacidad de cálculo de un ordenador son proporcionales al número de celdas de memoria y al número de procesadores respectivamente, es decir, al “tamaño” del ordenador. Entonces la capacidad de un ordenador clásico (de almacenamiento y de cálculo) crece linealmente con respecto a su tamaño.

En un ordenador cuántico la situación cambia por completo, hasta el punto que su capacidad crece exponencialmente con respecto a su tamaño. Este hecho, estrechamente relacionado con el principio de superposición de la Mecánica Cuántica, se denomina paralelismo cuántico. Llamamos qubits o bits cuánticos a los sistemas cuánticos elementales, es decir, a los sistemas cuánticos de dos estados. Los sistemas cuánticos de n qubits se describen mediante vectores de un espacio de Hilbert de dimensión 2^n .

*Este trabajo ha sido subvencionado por el MCYT, proyecto TIC2002-01541.

[†]Escuela U. y Facultad de Informática, U. Politécnica Madrid (<http://www.dma.eui.upm.es/seminarios/CompCuant>).

Esto permite codificar una cantidad exponencial de información en el estado de un sistema cuántico de n qubits. Además, cualquier transformación del estado del sistema se traduce en la modificación simultánea de toda la información almacenada. En definitiva, la capacidad de un ordenador cuántico (tanto de almacenamiento como de cálculo) crece exponencialmente con respecto a su tamaño.

Sin embargo, la medición de estados cuánticos es un inconveniente importante para la computación cuántica. Hay que recordar que las medidas cuánticas no son deterministas. Esto quiere decir, por ejemplo, que si medimos dos estados iguales los resultados no tienen por qué ser iguales. El proceso de medida es, por tanto, un experimento aleatorio en el que la probabilidad de cada resultado está determinada por el estado del sistema.

Las dificultades para sacar provecho del paralelismo cuántico son tan notables que hubo que esperar más de una década para encontrar el primer gran resultado. En 1994 Peter W. Shor sorprendió a todos presentando un algoritmo polinomial para factorizar números enteros [14]. Fue el primer problema en el que se alcanzaba una aceleración exponencial con respecto a los mejores algoritmos clásicos conocidos. Este descubrimiento generó gran actividad tanto en algorítmica cuántica como en criptografía cuántica.

Este resultado rompió teóricamente el sistema criptográfico más difundido en la actualidad, el sistema RSA propuesto por Rivest, Shamir y Adleman en 1978 [13], por lo que se empezaron a investigar sistemas criptográficos cuánticos. Las técnicas que se utilizan para garantizar la confidencialidad de los canales cuánticos se apoyan en una característica muy importante de los estados cuánticos. Se trata de la imposibilidad de copiar (clonar) estos estados. En el área de las comunicaciones, además del estudio de la confidencialidad, se están investigando otros problemas como, por ejemplo, la codificación de información clásica en canales cuánticos y el teletransporte de estados cuánticos.

Sin embargo el estudio de este modelo de computación apenas si ha comenzado. Hasta el momento sólo se ha podido hacer efectiva una ganancia exponencial en el cálculo de transformadas de Fourier y, en estos momentos, ésta es la herramienta más importante de la computación cuántica. Otra técnica que permite mejorar la complejidad de algunos algoritmos clásicos, aunque con ganancia solamente cuadrática, es el método de Grover de búsqueda en conjuntos no estructurados [9].

Los ordenadores cuánticos, a diferencia de los clásicos, son dispositivos analógicos. Esto podría parecer a priori un inconveniente grave, incluso insalvable. Basta considerar las enormes dificultades de desarrollo de la electrónica analógica frente a la electrónica digital para que este temor se acreciente. Ante esta perspectiva es evidente la necesidad de desarrollar una teoría de corrección de errores cuánticos. La investigación en esta línea, iniciada por Calderbank y Shor en 1996 [4] y por otros investigadores, ha demostrado que es posible corregir errores continuos en sistemas cuánticos.

Existen numerosos artículos de introducción a la computación cuántica entre los que destacamos los de Aharonov [1], Galindo y Martín-Delgado [8], Rieffel y Polak [12] y Steane [15]. Para profundizar más sobre el tema se puede ver, por ejemplo, el curso de Preskill [11]. Y si se quiere una fuente más completa el libro de Nielsen y Chuang [10] es una buena referencia.

El contenido del artículo está organizado del siguiente modo. En las secciones 2 y 3 se describen los modelos de computación clásico y cuántico respectivamente desde un punto de vista más práctico que teórico. El modelo cuántico se introduce a partir del modelo clásico para que se aprecien con claridad las analogías y las diferencias. En la sección 4 se presentan en detalle algunos de los algoritmos cuánticos más representativos y finalmente, en la sección 5, se comentan las expectativas de la computación cuántica.

2 Modelo clásico de computación

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 $V = \{0, 1\}$. En los ordenadores clásicos un bit se representa por el estado de carga de un condensador: si está descargado el bit vale 0 y si está cargado el bit vale 1. Los condensadores que representan los bits, junto con otros componentes electrónicos, se construyen en circuitos integrados.

Evidentemente un bit tiene poca información. Para representar cantidades mayores de información se

utilizan conjuntos de n bits que se llaman cadenas de bits. Por ejemplo 0110 es una cadena de 4 bits. Desde un punto de vista un poco más formal una cadena de n bits se puede considerar como un elemento del producto cartesiano $V^n = V \times \dots \times V$. Una cadena de bits puede representar cualquier información. Para ello basta establecer un mecanismo de codificación. Por ejemplo, una palabra de a lo sumo 5 caracteres y un número natural de a lo sumo 3 dígitos decimales se pueden representar en una cadena de 50 bits mediante el siguiente mecanismo de codificación:

- i) Cada letra se representa por 8 bits usando el código ASCII.
- ii) El número se representa por 10 bits usando el sistema de numeración binario.

Cuando una cadena de bits representa varios objetos, los bits asociados a cada uno de ellos (habitualmente consecutivos) se agrupan en registros. Un registro es por lo tanto un conjunto consecutivo de bits. En el ejemplo anterior el primer registro tiene 40 bits (para la palabra) y el segundo 10 bits (para el número). En la siguiente tabla se muestra la cadena de bits que representa la palabra “casas” y el número 588.

registro 1					registro 2
01100011	01100001	01110011	01100001	01110011	1001001100
<i>c</i>	<i>a</i>	<i>s</i>	<i>a</i>	<i>s</i>	588

En el modelo clásico de computación un algoritmo es un mecanismo para manipular cadenas de bits. Desde el punto de vista formal se puede considerar como un mecanismo para evaluar funciones booleanas. En efecto, 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 $V^n \rightarrow V^n$ tal que $f(\alpha) = \beta$ entonces el algoritmo es un mecanismo para evaluar f .

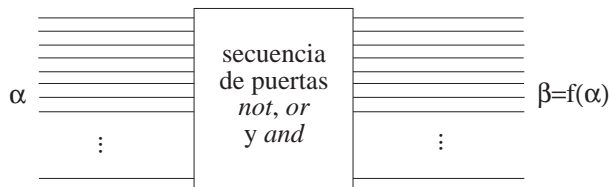


Figure 1: Algoritmos en el modelo clásico de computación

En un algoritmo hay que detallar el mecanismo de manipulación de la cadena de bits hasta reducirlo a una secuencia de puertas lógicas (véase la figura 1). El motivo es perfectamente comprensible: los ordenadores clásicos sólo son capaces de evaluar puertas lógicas, no son capaces de evaluar funciones booleanas genéricas. Es un resultado de sobra conocido que las puertas lógicas *not*, *or* y *and* descritas en la figura 2 permiten definir cualquier función booleana.

x	<i>not</i> x
0	1
1	0

x	y	<i>x or</i> y
0	0	0
0	1	1
1	0	1
1	1	1

x	y	<i>x and</i> y
0	0	0
0	1	0
1	0	0
1	1	1

Figure 2: Definición de las puertas lógicas *not*, *or* y *and*

La restricción que hemos impuesto a los algoritmos, exigiendo que las cadenas de bits de entrada y salida tengan la misma longitud, se puede suprimir sin afectar apenas al modelo de computación. Hemos decidido mantenerla para recalcar más las analogías entre los modelos de computación clásico y cuántico. Para describir algoritmos, en términos de puertas lógicas, se utilizan diagramas como los de la figura 3.

A continuación mostramos, como ejemplo, un algoritmo para sumar. Los sumandos se representan en registros de n bits mientras que la suma se obtiene en un registro de $n + 1$ bits. Por supuesto, tanto los sumandos como la suma se codifican en el sistema de numeración binario. En la figura 4 se muestra el algoritmo para $n = 2$, el algoritmo general se obtiene como una generalización inmediata de éste.

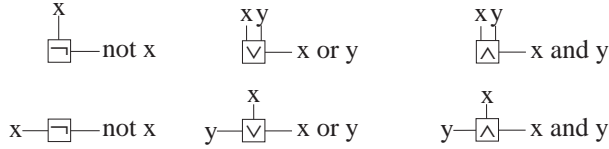


Figure 3: Diagramas de las puertas lógicas *not*, *or* y *and*

El hardware actual puede realizar muchas puertas lógicas a la vez. Por ejemplo, puede sumar dos enteros de 64 bits en unos pocos pasos de reloj. Por el contrario, nuestro modelo sólo puede realizar una puerta lógica por paso. Es evidente que, a pesar de esto, el modelo que hemos descrito es equivalente al modelo real en cuanto a capacidad de cálculo.

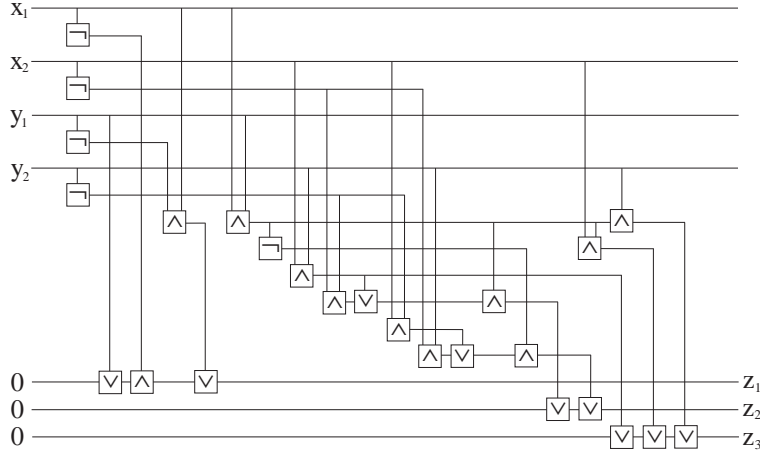


Figure 4: Algoritmo clásico para sumar: $x_2x_1 + y_2y_1 = z_3z_2z_1$

3 Modelo cuántico de computación

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 conocido por los profanos en la materia es, sin duda, el spin de un electrón. En este sistema podemos representar el spin $-\frac{1}{2}$ por el estado $|0\rangle$ y el spin $+\frac{1}{2}$ por el estado $|1\rangle$.

Hasta ahora el modelo cuántico no se diferencia del clásico: 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

$$\Psi = \frac{1}{2}|0\rangle + \frac{\sqrt{3}}{2}|1\rangle \quad (1)$$

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 $\mathcal{V} = L(|0\rangle, |1\rangle)$. Según la Mecánica Cuántica \mathcal{V} es un espacio de Hilbert complejo en el que $\mathcal{B} = [|0\rangle, |1\rangle]$ es una base ortonormal y los estados son vectores unitarios.

Entonces un qubit puede estar en cualquier estado $\Psi = a|0\rangle + b|1\rangle$ tal que $a, b \in \mathcal{C}$ y $|a|^2 + |b|^2 = 1$. Los coeficientes a y b de Ψ 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, por ejemplo el de la expresión (1). 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, es decir, que sólo puede

tener los valores $-\frac{1}{2}$ y $+\frac{1}{2}$. Además estos valores corresponden a los estados $|0\rangle$ y $|1\rangle$ respectivamente. Por lo tanto el estado intermedio Ψ no tiene spin definido.

Como consecuencia de esta situación surge la segunda gran diferencia entre los modelos de computación clásico y cuántico. Siempre es posible medir (leer) el valor de un bit pero generalmente no es posible medir el estado de un qubit. Entonces ¿qué información proporciona la medida de un qubit?

Para entenderlo, sigamos con el ejemplo del spin de un electrón. Vamos a estudiar lo que ocurre al medir el qubit definido en la expresión (1). Para ello empleamos el dispositivo esquematizado en la figura 5. 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, hacia abajo si su spin es $-\frac{1}{2}$ y hacia arriba si su spin es $+\frac{1}{2}$. Finalmente el electrón atraviesa una de las dos rendijas del panel 2, la inferior si su spin es $-\frac{1}{2}$ y la superior si su spin es $+\frac{1}{2}$.

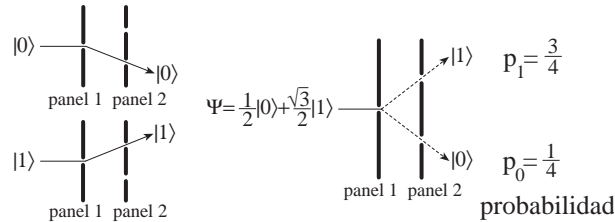


Figure 5: Medida del spin del estado intermedio $\Psi = \frac{1}{2}|0\rangle + \frac{\sqrt{3}}{2}|1\rangle$

El qubit Ψ que estamos analizando es un estado intermedio y en consecuencia no tiene spin definido. No es difícil convencerse de que 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 las sorpresas todavía no han acabado. Si el electrón ha pasado por la rendija inferior su spin, después de la medida, sólo puede ser $-\frac{1}{2}$ 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 $+\frac{1}{2}$ 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 (p_1) de que el estado final del qubit sea $|0\rangle$ ($|1\rangle$) es igual al cuadrado del módulo de la amplitud de $|0\rangle$ ($|1\rangle$) en la combinación lineal. Para el qubit Ψ del ejemplo el resultado final será $|0\rangle$ con probabilidad $p_0 = \frac{1}{4}$ y $|1\rangle$ con probabilidad $p_1 = \frac{3}{4}$.

En la tabla de la figura 6 se resume el proceso de medida de un qubit. Diremos que el resultado de la medida es 0 (1) si el estado final es $|0\rangle$ ($|1\rangle$).

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	$\frac{b}{ b } 1\rangle$	$p_1 = b ^2$

Figure 6: Medida de un qubit

3.1 Sistemas de 2-qubits

La información que contiene un qubit es evidentemente muy pequeña. Para poder representar cantidades mayores de información se recurre a sistemas de n -qubits. Para empezar con un ejemplo sencillo supon-

gamos 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.

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

$$\Psi = \frac{1}{4}|00\rangle + \frac{\sqrt{3}}{4}|01\rangle + \frac{\sqrt{3}}{4}|10\rangle + \frac{3}{4}|11\rangle \quad (2)$$

Entonces un 2–qubit es un vector del espacio vectorial $\mathcal{V}_2 = L(|00\rangle, |01\rangle, |10\rangle, |11\rangle)$ que, según la Mecánica Cuántica, es un espacio de Hilbert complejo en el que $\mathcal{B}_2 = [|00\rangle, |01\rangle, |10\rangle, |11\rangle]$ es una base ortonormal y los estados son vectores unitarios. De este modo, un 2–qubit puede estar en cualquier estado de la forma $\Psi = a|00\rangle + b|01\rangle + c|10\rangle + d|11\rangle$ tal que $a, b, c, d \in \mathcal{C}$ y $|a|^2 + |b|^2 + |c|^2 + |d|^2 = 1$.

La descripción que hemos hecho de los sistemas de 2–qubits es completa. Sin embargo no sabemos qué relación existe entre el espacio vectorial \mathcal{V}_2 y los espacios vectoriales \mathcal{V} asociados a los dos qubits, considerados como sistemas independientes. Como cabía esperar, \mathcal{V}_2 es el producto tensorial $\mathcal{V} \otimes \mathcal{V}$. Del mismo modo los vectores de la base \mathcal{B}_2 corresponden a los distintos productos tensoriales de los vectores de \mathcal{B} :

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

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

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

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

$$\Psi = \frac{1}{\sqrt{3}}|00\rangle + \frac{1}{\sqrt{3}}|01\rangle + \frac{1}{\sqrt{3}}|10\rangle \quad (3)$$

Es sencillo darse cuenta de que, 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 \mathbb{R}^8) la probabilidad de que el resultado sea un estado entrelazado es 1. Para entender mejor estos estados, vamos a caracterizarlos mediante una propiedad menos formal. Para ello necesitamos generalizar el proceso de medida.

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. Y tomemos como ejemplo el estado dado en la expresión (3). 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 estado

$$|0\rangle \otimes (a|0\rangle + b|1\rangle) = a|00\rangle + b|01\rangle \quad \text{ó} \quad |1\rangle \otimes (a|0\rangle + b|1\rangle) = a|10\rangle + b|11\rangle$$

para $a, b \in \mathcal{C}$ 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.

$$\Psi = \frac{\sqrt{2}}{\sqrt{3}}\Psi_0 + \frac{1}{\sqrt{3}}\Psi_1 \quad \text{donde} \quad \Psi_0 = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|01\rangle \quad \text{y} \quad \Psi_1 = |10\rangle \quad (4)$$

A la vista de la expresión (4), resulta bastante natural suponer que si el resultado de la medida es 0 el estado final será Ψ_0 y que si es 1 el estado final será Ψ_1 . En efecto así es, y además la probabilidad p_0 (p_1) de que la medida sea 0 (1) es igual al módulo al cuadrado del coeficiente de Ψ_0 (Ψ_1). En tabla de la figura 7 se resume el proceso de medida.

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	$\Psi_1 = 10\rangle$	$p_1 = \frac{1}{3}$

Figure 7: Medida del primer qubit

La probabilidad p_0 y el estado final 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.

Como ya se había comentado, los estados entrelazados se pueden caracterizar a partir de las medidas. Sea Ψ un 2-qubit en el que medimos el primer qubit y a continuación, en el estado resultante, medimos el segundo qubit. Entonces Ψ es un estado entrelazado si y sólo si la probabilidad de que la medida del segundo qubit sea 0 ó 1 depende del resultado de la medida del primer qubit. Tomemos como ejemplo el 2-qubit dado en la expresión (3) que es un estado entrelazado. En la tabla 8, donde aparece resumido el proceso de medida de los dos qubits, se puede comprobar la caracterización anterior.

Medida 1	Estado final 1	Medida 2	Prob. de la medida 2
0	$\Psi_0 = \frac{1}{\sqrt{2}} 00\rangle + \frac{1}{\sqrt{2}} 01\rangle$	0	$p_0 = \frac{1}{2}$
0	$\Psi_0 = \frac{1}{\sqrt{2}} 00\rangle + \frac{1}{\sqrt{2}} 01\rangle$	1	$p_1 = \frac{1}{2}$
1	$\Psi_1 = 10\rangle$	0	$p_0 = 1$
1	$\Psi_1 = 10\rangle$	1	$p_1 = 0$

Figure 8: Medida de los dos qubits del estado $\frac{1}{\sqrt{3}}|00\rangle + \frac{1}{\sqrt{3}}|01\rangle + \frac{1}{\sqrt{3}}|10\rangle$

Ejercicios:

- Demostrar que para todo 2-qubit Ψ existen α, β, Ψ_0 y Ψ_1 únicos tales que $\alpha, \beta \in \mathbb{R}$, Ψ_0 y Ψ_1 son 2-qubits pertenecientes a $L(|00\rangle, |01\rangle)$ y $L(|10\rangle, |11\rangle)$ respectivamente y $\Psi = \alpha\Psi_0 + \beta\Psi_1$.
- Sea Ψ un 2-qubit en el que medimos el primer qubit y, a continuación, el segundo qubit y sea p^0 (p^1) la probabilidad de que la segunda medida sea 0 si la primera fue 0 (1). Demostrar que Ψ es un estado entrelazado si y sólo si $p^0 \neq p^1$.

3.2 Sistemas de n-qubits

La generalización de 2-qubits a n -qubits resulta ahora muy sencilla. Un n -qubit es un vector unitario del espacio de Hilbert complejo $\mathcal{V}_n = \mathcal{V} \otimes \dots \otimes \mathcal{V}$ en el que $\mathcal{B}_n = [|0\dots 00\rangle, |0\dots 01\rangle, |0\dots 10\rangle, \dots, |1\dots 11\rangle]$ es una base ortonormal, llamada base computacional. Un vector genérico de la base \mathcal{B}_n se puede ver como un producto tensorial del siguiente modo

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

La cadena de bits $x_1 x_2 \dots x_n$ la podemos interpretar como un número natural x representado en el sistema de numeración binario. De este modo los vectores de la base \mathcal{B}_n se identifican con los números

naturales x que cumplen $0 \leq x < 2^n$ (números con n dígitos binarios). Y, una vez identificada la cadena de bits $x_1x_2 \dots x_n$ con el número natural x , se puede escribir x en el sistema de numeración decimal. En definitiva podemos escribir $\mathcal{B}_n = [|0\rangle, |1\rangle, |2\rangle, \dots, |2^n - 1\rangle]$.

La identificación de los vectores de la base \mathcal{V}_n 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

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

Conviene resaltar que la dimensión de \mathcal{V}_n es exponencial, concretamente 2^n . Ésta es la propiedad clave del denominado paralelismo cuántico. Por el momento sólo podemos apreciar 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 el 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 $|x\rangle, |y\rangle, |z\rangle \dots$ donde los números $x, y, z \dots$ son números con k dígitos binarios. En el ejemplo anterior, llamando $|x\rangle$ al registro completo de 16 qubits, $|y\rangle$ al registro de los 8 primeros qubits y $|z\rangle$ al registro de los 8 últimos qubits, los estados Ψ_1 y Ψ_2 se pueden escribir del siguiente modo

$$\Psi_1 = |10000011\rangle \otimes |11010011\rangle = |131\rangle \otimes |211\rangle \quad \text{y} \quad \Psi_2 = \frac{1}{256} \sum_{y=0}^{255} \sum_{z=0}^{255} |y\rangle \otimes |z\rangle$$

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 descrito para 2-qubits y está descrito en la tabla de la figura 9.

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

Figure 9: Medida del k -ésimo qubit

3.3 Algoritmos cuánticos

En el modelo cuántico de computación un algoritmo es un mecanismo para manipular n -qubits. Ya conocemos uno de los dos posibles mecanismos para hacerlo: medir qubits. El otro consiste en transformar un estado inicial Ψ_1 en su correspondiente estado final Ψ_2 . Si llamamos U a la función de $\mathcal{V}_n \rightarrow \mathcal{V}_n$ tal que

$U\Psi_1 = \Psi_2$ entonces el segundo mecanismo consiste en aplicar la función U . 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.

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, tal como está esquematizado en la figura 10.

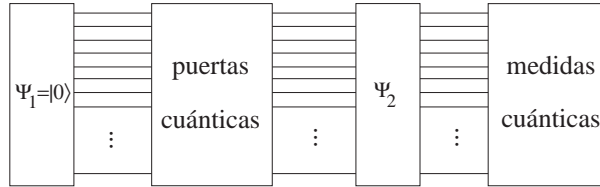


Figure 10: Algoritmos en el modelo cuántico de computación

Obsérvese que 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.

La tercera y última gran diferencia entre los modelos de computación cuántico y clásico está relacionada con las puertas. Mientras que las puertas cuánticas son biyectivas, al ser transformaciones unitarias, las puertas lógicas en general no lo son. Por ejemplo, la siguiente transformación clásica de dos bits no lo es:

$$\boxed{x \mid y} \quad \longrightarrow \quad \boxed{x \mid x \text{ and } y}$$

La consecuencia más importante de esta propiedad de las puertas cuánticas es que los estados cuánticos no se pueden copiar [6, 16]. Para copiar un n -qubit Ψ bastaría encontrar una transformación unitaria U que cumpliera $U(|x\rangle \otimes |0\rangle) = |x\rangle \otimes |x\rangle$ para todo $0 \leq x < 2^n$, donde los dos registros tienen n qubits. En efecto, dicha transformación cumpliría $U(\Psi \otimes |0\rangle) = \Psi \otimes \Psi$. Sin embargo la transformación U no existe como se prueba en el siguiente teorema.

Teorema 1 *No existe ninguna transformación unitaria $U : \mathcal{V}_{2n} \rightarrow \mathcal{V}_{2n}$ tal que $U(\Psi \otimes |0\rangle) = \Psi \otimes \Psi$ para todo n -qubit Ψ .*

Demostración: Sea U una transformación unitaria $U : \mathcal{V}_{2n} \rightarrow \mathcal{V}_{2n}$ tal que $U(|a\rangle \otimes |0\rangle) = |a\rangle \otimes |a\rangle$ y $U(|b\rangle \otimes |0\rangle) = |b\rangle \otimes |b\rangle$ tal que $a \neq b$ y $0 \leq a, b < 2^n$.

Consideremos el n -qubit $\Psi = \frac{1}{\sqrt{2}}(|a\rangle + |b\rangle)$. Entonces

$$U(\Psi \otimes |0\rangle) = \frac{1}{\sqrt{2}} (U(|a\rangle \otimes |0\rangle) + U(|b\rangle \otimes |0\rangle)) = \frac{1}{\sqrt{2}} (|a\rangle \otimes |a\rangle + |b\rangle \otimes |b\rangle) \neq \Psi \otimes \Psi$$

En definitiva, ninguna transformación unitaria es capaz de copiar los estados $|a\rangle$, $|b\rangle$ y Ψ . ■

Otra estrategia posible para copiar un estado consiste en medirlo y, a continuación, reproducirlo. Esta estrategia tropieza con el hecho, expuesto anteriormente, de que la medida de un estado no permite conocerlo completamente y, en consecuencia, no aporta información suficiente para reproducirlo.

Ejercicios:

- Sea $\Psi = a|0\rangle + b|1\rangle$ un qubit. Demostrar que existe una transformación unitaria $U : \mathcal{V}_2 \rightarrow \mathcal{V}_2$ tal que $U(\Psi \otimes |0\rangle) = a|00\rangle + b|11\rangle$.

- Supongamos que sabemos que el estado Ψ es igual a $|0\rangle$ o a $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$. Demostrar que no es posible determinar con certeza de cuál de los dos estados se trata.

3.4 Puertas cuánticas

Las puertas cuánticas más simples son las puertas de un solo qubit, es decir, transformaciones unitarias $U : \mathcal{V} \rightarrow \mathcal{V}$. Las más importantes, por su utilidad en el diseño de algoritmos, son las siguientes:

- La transformación H de Hadamard:
$$\begin{cases} H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \\ H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \end{cases}, \text{ es decir, } H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$
- La identidad I :
$$\begin{cases} I|0\rangle = |0\rangle \\ I|1\rangle = |1\rangle \end{cases}, \text{ es decir, } I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$
- La negación X :
$$\begin{cases} X|0\rangle = |1\rangle \\ X|1\rangle = |0\rangle \end{cases}, \text{ es decir, } X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$
- El cambio de fase Z :
$$\begin{cases} Z|0\rangle = |0\rangle \\ Z|1\rangle = -|1\rangle \end{cases}, \text{ es decir, } Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$
- La negación y cambio de fase Y :
$$\begin{cases} Y|0\rangle = -|1\rangle \\ Y|1\rangle = |0\rangle \end{cases}, \text{ es decir, } Y = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

Las matrices de las transformaciones I , X , $-iY$ y Z se conocen con el nombre de matrices de Pauli. Generan un grupo multiplicativo de orden 8 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 X .

Para construir algoritmos cuánticos no son suficientes las puertas de un qubit. Se necesitan además puertas de dos qubits. La más importante es la puerta denominada *Cnot* (Controlled-Not).

$$Cnot : \begin{cases} |0x\rangle \rightarrow |0x\rangle \\ |1x\rangle \rightarrow |1\rangle \otimes X|x\rangle \end{cases}, \text{ es decir, } Cnot = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \text{ ó } Cnot = \begin{pmatrix} I & 0 \\ 0 & X \end{pmatrix}$$

La transformación *Cnot* aplica la negación X al segundo qubit si el primero está en estado $|1\rangle$. Lo mismo se puede hacer sustituyendo la puerta X por una puerta cualquiera de un qubit, U . De este modo obtenemos la puerta denominada *CU* (Controlled- U). Obsérvese que la puerta *Cnot* es un caso particular, es decir, $Cnot = CX$.

$$CU : \begin{cases} |0x\rangle \rightarrow |0x\rangle \\ |1x\rangle \rightarrow |1\rangle \otimes U|x\rangle \end{cases}, \text{ es decir, } CU = \begin{pmatrix} I & 0 \\ 0 & U \end{pmatrix}$$

Un conjunto de puertas cuánticas se llama universal si permite aproximar cualquier transformación unitaria, con una cota de error dada a priori, mediante una secuencia finita de puertas del conjunto. Barenco y otros [2] probaron que el conjunto de puertas cuánticas $\mathcal{P} = \{Cnot\} \cup \{U \mid U \text{ es de 1 qubit}\}$ es universal. Pero, aunque el conjunto \mathcal{P} es universal, con frecuencia se usan puertas cuánticas que no pertenecen a \mathcal{P} . Una de las más frecuentes es una puerta de 3 qubits, T , llamada puerta de Toffoli. Es una generalización de la puerta *Cnot*, pues aplica la negación X al tercer qubit si los dos primeros están en estado $|1\rangle$.

$$T : \begin{array}{l} |00x\rangle \rightarrow |00x\rangle \\ |01x\rangle \rightarrow |01x\rangle \\ |10x\rangle \rightarrow |10x\rangle \\ |11x\rangle \rightarrow |11\rangle \otimes X|x\rangle \end{array}, \text{ es decir, } T = \begin{pmatrix} I & 0 & 0 & 0 \\ 0 & I & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & X \end{pmatrix}$$

La puerta de Toffoli, además de su utilidad en el diseño de algoritmos cuánticos, permite realizar cualquier computación clásica. Esto significa que cualquier algoritmo clásico se puede simular mediante un algoritmo cuántico que solamente usa la puerta de Toffoli. Para demostrarlo basta probar que, con la puerta de Toffoli, se pueden simular las puertas lógicas *not* y *and*.

$$\left. \begin{array}{l} T : |11x\rangle \longrightarrow |11\rangle \otimes X|x\rangle \\ T : |xy0\rangle \longrightarrow |xy0\rangle \quad \text{si } \text{not}(x \text{ and } y) \\ T : |xy0\rangle \longrightarrow |xy1\rangle \quad \text{si } x \text{ and } y \end{array} \right\} \begin{array}{l} \boxed{1} \boxed{1} \boxed{x} \longrightarrow \boxed{1} \boxed{1} \boxed{\text{not } x} \\ \boxed{x} \boxed{y} \boxed{0} \longrightarrow \boxed{x} \boxed{y} \boxed{x \text{ and } y} \end{array}$$

Es evidente que, en un sistema de n qubits, hay que especificar los qubits sobre los que actúan las puertas cuánticas. Si U es una puerta de un qubit escribiremos $U(i)$ para indicar que U se aplica al i -ésimo qubit, y escribiremos $CU(i, j)$ para indicar que U se aplica al j -ésimo qubit si el i -ésimo qubit está en estado $|1\rangle$. Del mismo modo escribiremos $T(i, j, k)$ para indicar que X se aplica al k -ésimo qubit si los qubits i -ésimo y j -ésimo están en estado $|1\rangle$. A continuación se muestran, a modo de ejemplo, las matrices de las puertas $X(2)$ y $Cnot(2, 1)$ en un sistema de 2 qubits.

$$X(2) = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad Cnot(2, 1) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

Analicemos un poco más a fondo la puerta cuántica $X(2)$ en un sistema de 2 qubits. Lo primero que observamos es que en su matriz aparecen dos copias de la matriz de X , considerada como puerta en un sistema de 1 qubit. Es fácil comprobar que en un sistema de n qubits el número de copias sería 2^{n-1} . Para las puertas de dos qubits $Cnot(i, j)$ el número copias de X es 2^{n-2} y para la puerta de Toffoli el número de copias de X es 2^{n-3} . Aquí podemos apreciar claramente la capacidad exponencial de cálculo de los sistemas cuánticos que, junto con la capacidad exponencial de almacenamiento de información ya comentada, constituyen el denominado paralelismo cuántico.

Volvamos a la puerta cuántica $X(2)$ en un sistema de 2 qubits. Evidentemente se trata de una transformación unitaria sobre \mathcal{V}_2 y, sin embargo, X sólo está definida sobre \mathcal{V} . La explicación es muy sencilla: la transformación X actúa sobre el segundo factor del producto $\mathcal{V}_2 = \mathcal{V} \otimes \mathcal{V}$, induciendo una transformación unitaria sobre \mathcal{V}_2 . La transformación unitaria inducida es, desde un punto de vista un poco más formal, el producto tensorial $I \otimes X$.

Para terminar introducimos la transformada de Walsh-Hadamard: $W = H \otimes \dots \otimes H$. Se trata de la composición de n puertas H , una sobre cada qubit. La propiedad más importante de esta transformación es que aplica el primer vector de la base \mathcal{B}_n en una superposición de todos los vectores de la base:

$$W|0\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle$$

Se cumple en general que $W|y\rangle$, para todo $0 \leq y < 2^n$, es una superposición de todos los vectores de la base. El valor de $W|y\rangle$ se puede expresar como

$$W|y\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} (-1)^{x \cdot y} |x\rangle \quad (5)$$

donde $x \cdot y = x_1 y_1 \oplus \dots \oplus x_n y_n$, siendo $x_n \dots x_1$ y $y_n \dots y_1$ las representaciones binarias de los números enteros x e y y \oplus la suma módulo 2.

Ejercicios:

- Determinar una secuencia de puertas y medidas cuánticas que transforme el qubit $|\Psi\rangle$ en $|0\rangle$.
Nota: Se pueden aplicar puertas distintas dependiendo del resultado de una medida.
- Se define la puerta de dos qubits $Swap|xy\rangle = |yx\rangle$. Expresar la puerta $Swap$ como una secuencia de puertas $Cnot$.
- Encontrar una secuencia de puertas que transforme $|0000\rangle \rightarrow \frac{1}{2}(|00\rangle + |11\rangle) \otimes (|00\rangle + |11\rangle)$ y $|1000\rangle \rightarrow \frac{1}{2}(|00\rangle - |11\rangle) \otimes (|00\rangle - |11\rangle)$.
- Demostrar que la transformada de Walsh-Hadamard se obtiene como se indica en la fórmula 5.

3.5 Representación de puertas y algoritmos cuánticos

Los algoritmos cuánticos, igual que los clásicos, se pueden representar mediante diagramas. En la figura 11 se muestran los distintos símbolos que se utilizan para definir algoritmos cuánticos. Los qubits se representan por líneas horizontales, se numeran verticalmente empezando desde arriba y evolucionan temporalmente de izquierda a derecha. Los estados inicial y final de un qubit se pueden especificar escribiéndolos a la izquierda y a la derecha respectivamente de la línea correspondiente al qubit.

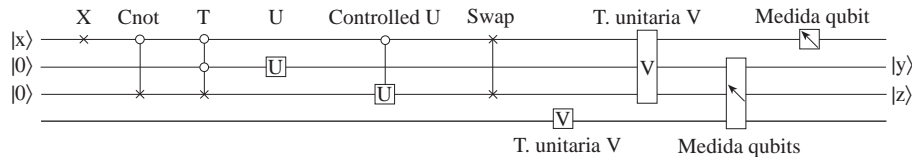


Figure 11: Diagramas de algoritmos cuánticos

La puerta X de un qubit se representa colocando el símbolo \times sobre el qubit y, en general, la puerta U de un qubit se especifica mediante el símbolo \boxed{U} . Para representar la puerta $Cnot$ se unen con un segmento vertical los símbolos \times y \circ colocados sobre el qubit afectado y el qubit de control respectivamente y, en general, la puerta CU (controlled-U) se especifica sustituyendo \times por \boxed{U} en la puerta $Cnot$.

La puerta de Toffoli T se representa uniendo con un segmento vertical los símbolos \times , \circ y \circ colocados sobre el qubit afectado y los dos qubits de control respectivamente. La puerta $Swap$ se especifica uniendo los símbolos \times y \times colocados sobre los dos qubits afectados.

Con frecuencia se utilizan líneas horizontales para representar registros de más de un qubit. También se puede especificar una transformación unitaria V sobre uno o varios registros, colocando el símbolo \boxed{V} sobre ellos. Finalmente la medida de uno o varios qubits se indica colocando el símbolo $\boxed{\nabla}$ sobre ellos.

A modo de ejemplo mostramos un algoritmo cuántico para sumar. Los sumandos son números de n dígitos binarios mientras que la suma se obtiene como un número de $n + 1$ dígitos. En la figura 12 se muestra el algoritmo para $n = 2$, el algoritmo general se obtiene como una generalización sencilla de éste.

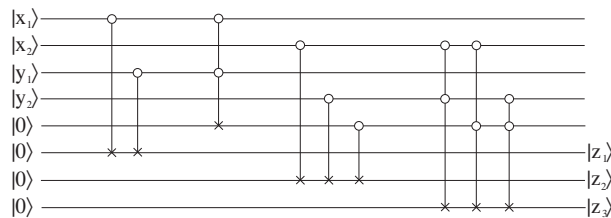


Figure 12: Algoritmo cuántico para sumar: $x_2x_1 + y_2y_1 = z_3z_2z_1$

4 Algoritmos cuánticos relevantes

Los primeros problemas que se abordaron en computación cuántica, aunque sencillos, permitieron comprobar en la práctica las diferencias fundamentales entre los modelos de computación clásico y cuántico. También aportaron técnicas algorítmicas que permitirían posteriormente obtener algunos algoritmos importantes como el de Shor [14] y el de Grover [9]. Dado el carácter introductorio de este artículo limitaremos nuestro estudio al análisis de los primeros algoritmos cuánticos.

Los primeros problemas que se plantearon en computación cuántica consistieron en el estudio de propiedades de funciones booleanas. Es en este tipo de problemas donde la computación cuántica manifiesta la potencia del paralelismo cuántico, consiguiendo mejoras exponenciales respecto de los algoritmos clásicos. Antes de entrar en la descripción de los problemas y de resolverlos vamos a ver cómo trabajar con funciones booleanas.

Generalmente la información se codifica en la cadena de bits que identifica a los distintos estados de la base de computación. Sea $f : \{0, 1\}^n \rightarrow \{0, 1\}^m$ una función booleana genérica. La forma más habitual para representar f , de forma que se pueda evaluar, en un $(n + m)$ -qubit es

$$|x\rangle \otimes |0\rangle \longrightarrow |x\rangle \otimes |f(x)\rangle \quad (6)$$

Es importante hacer notar que se representa el grafo de la función $(x, f(x))$, en lugar de $f(x)$, porque debe ser una función inyectiva. Se puede comprobar fácilmente que la transformación $|x\rangle \rightarrow |f(x)\rangle$ es unitaria si y sólo si f es biyectiva y, en particular, es necesario que $n = m$. La definición de la expresión (6) no define ninguna transformación unitaria, pero puede extenderse para que así sea. Una forma sencilla para extender la definición (6) a una transformación unitaria es

$$U_f (|x\rangle \otimes |y\rangle) = |x\rangle \otimes |y \oplus f(x)\rangle$$

donde \oplus es la suma módulo 2 de cadenas de bits, en este caso de longitud m . Es sencillo probar que U es una biyección de los estados de la base computacional. Para ello es suficiente demostrar que U es inyectiva. Si $|x\rangle \otimes |y \oplus f(x)\rangle = |x'\rangle \otimes |y' \oplus f(x')\rangle$ entonces $x = x'$ e $y \oplus f(x) = y' \oplus f(x')$. Entonces $x = x'$ e $y = y' \oplus f(x') \oplus f(x) = y' \oplus 0 = y'$ y por tanto U es inyectiva. Diremos que U_f es la transformación unitaria asociada a la función booleana f .

El complemento de la función booleana $f : \{0, 1\}^n \rightarrow \{0, 1\}$ se denota por \bar{f} . De este modo si $f(x) = 0$ entonces $\bar{f}(x) = 1$, mientras que si $f(x) = 1$ entonces se cumple $\bar{f}(x) = 0$.

Problema 1 (*Problema de Deutsch*) Dada una función booleana $f : \{0, 1\} \rightarrow \{0, 1\}$ y su transformación unitaria asociada U_f , encontrar un algoritmo que determine si f es constante o no aplicando U_f el menor número posible de veces.

Es obvio que clásicamente hay que evaluar $f(0)$ y $f(1)$ para resolver el problema planteado. Sin embargo, cuánticamente sólo es preciso evaluar U_f una vez. Esto se puede conseguir gracias al paralelismo cuántico que permite evaluar simultáneamente $f(0)$ y $f(1)$:

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

Aunque generalmente los algoritmos cuánticos son probabilísticos, en este caso es posible obtener un algoritmo determinista para resolver el problema. Generalmente es fácil utilizar el paralelismo cuántico, lo más difícil es conseguir que la probabilidad de obtener el resultado buscado sea grande. Vamos a ver cómo se consigue en el problema de Deutsch que que dicha probabilidad sea igual a 1.

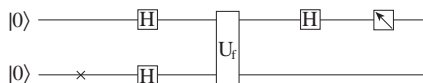


Figure 13: Circuito para el problema de Deutsch

A continuación se muestra el estado del 2-qubit después de aplicar cada una de las transformaciones unitarias del circuito y el resultado de la medida final. Llamamos α a una variable booleana que nos indica si f es una función constante tomando el valor 0 o si no lo es tomando el valor 1.

$$\begin{aligned}
|0\rangle \otimes |0\rangle &\longrightarrow |0\rangle \otimes |1\rangle \\
&\longrightarrow \frac{1}{2} (|0\rangle + |1\rangle) \otimes (|0\rangle - |1\rangle) \\
&\longrightarrow \frac{1}{2} (|0\rangle \otimes (|f(0)\rangle - |\bar{f}(0)\rangle) + |1\rangle \otimes (|f(1)\rangle - |\bar{f}(1)\rangle)) \\
&= \frac{1}{2} (|0\rangle + (-1)^\alpha |1\rangle) \otimes (|f(0)\rangle - |\bar{f}(0)\rangle) \\
&\longrightarrow \frac{1}{\sqrt{2}} |\alpha\rangle \otimes (|f(0)\rangle - |\bar{f}(0)\rangle) \\
&\longrightarrow \alpha \quad (\text{probabilidad} = 1)
\end{aligned}$$

Por tratarse de un problema de complejidad constante, es decir, que no depende de un parámetro $n \in \mathbb{N}$, no podemos sacar ninguna conclusión al compararlo con los algoritmos clásicos. Para poder hacerlo vamos a introducir una generalización del problema. Se dice que una función booleana $f : \{0, 1\}^n \rightarrow \{0, 1\}$ es balanceada si toma el valor 0 el mismo número de veces que el valor 1.

Problema 2 (*Problema de Deutsch-Jozsa*) Dada una función booleana $f : \{0, 1\}^n \rightarrow \{0, 1\}$ constante o balanceada y su transformación unitaria asociada U_f , encontrar un algoritmo que determine si f es constante o es balanceada aplicando U_f el menor número posible de veces.

Es sencillo probar que clásicamente hay que evaluar f sobre la mitad más uno de los elementos del dominio, es decir sobre $2^{n-1} + 1$ elementos, para resolver el problema planteado de forma determinista. Sin embargo, cuánticamente basta evaluar U_f una sola vez para obtener la solución determinista. Por tanto en la resolución de este problema el modelo cuántico de computación permite una ganancia exponencial respecto al modelo clásico. Este es el primer problema en el que se ha conseguido un aprovechamiento óptimo del paralelismo cuántico.

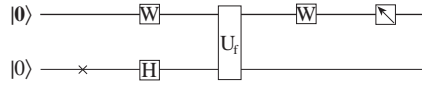


Figure 14: Circuito para el problema de Deutsch-Jozsa

A continuación se muestra el estado del $(n+1)$ -qubit después de aplicar cada una de las transformaciones unitarias del circuito. El resultado final se puede separar en dos sumandos, atendiendo al estado del n -qubit: en el primero está en estado $|0\rangle$ y en el segundo es ortogonal al estado $|0\rangle$.

$$\begin{aligned}
|0\rangle \otimes |0\rangle &\longrightarrow |0\rangle \otimes |1\rangle \\
&\longrightarrow \frac{1}{\sqrt{2^{n+1}}} \sum_{0 \leq x < 2^n} (|x\rangle \otimes (|0\rangle - |1\rangle)) \\
&\longrightarrow \frac{1}{\sqrt{2^{n+1}}} \sum_{0 \leq x < 2^n} (|x\rangle \otimes (|f(x)\rangle - |\bar{f}(x)\rangle)) \\
&\longrightarrow \frac{1}{2^n \sqrt{2}} \sum_{0 \leq x < 2^n} \sum_{0 \leq y < 2^n} ((-1)^{x \cdot y} |y\rangle \otimes (|f(x)\rangle - |\bar{f}(x)\rangle)) \\
&= \frac{1}{2^n \sqrt{2}} |0\rangle \otimes \sum_{0 \leq x < 2^n} (|f(x)\rangle - |\bar{f}(x)\rangle) + \\
&\quad \frac{1}{2^n \sqrt{2}} \sum_{0 < y < 2^n} \left(|y\rangle \otimes \sum_{0 \leq x < 2^n} ((-1)^{x \cdot y} (|f(x)\rangle - |\bar{f}(x)\rangle)) \right)
\end{aligned}$$

Analicemos el resultado anterior suponiendo que f es constante y que es balanceada. Si es constante el estado final del n -qubit es $|0\rangle$ y si es balanceada es ortogonal a $|0\rangle$. Por lo tanto la medida del n -qubit permite determinar de forma determinista si f es constante o balanceada.

$$\begin{aligned} \text{constante: } & \frac{1}{\sqrt{2}}|0\rangle \otimes (|f(0)\rangle - |\bar{f}(0)\rangle) \longrightarrow 0 \\ \text{balanceada: } & \frac{1}{2^n\sqrt{2}} \sum_{0 < y < 2^n} \left(|y\rangle \otimes \sum_{0 \leq x < 2^n} ((-1)^{x \cdot y} (|f(x)\rangle - |\bar{f}(x)\rangle)) \right) \longrightarrow \neq 0 \end{aligned}$$

Si sólo se busca una solución probabilística entonces existen algoritmos clásicos que evalúan la función $O(1)$ veces, donde la constante que esconde la O depende de la cota de error ϵ . Basta elegir aleatoriamente $O(1)$ elementos distintos del dominio de f y evaluar la función en los mismos. Si se obtienen valores distintos f es con seguridad balanceada mientras que si todos los valores son iguales f es probablemente constante, con un margen de error ϵ .

El último problema que a vamos a analizar no admite solución determinista en tiempo polinomial, ni en computación clásica ni en computación cuántica. Y sólo tiene solución probabilística polinomial en computación cuántica. Se dice que una función booleana $f : \{0, 1\}^n \rightarrow \{0, 1\}^n$ es periódica de periodo $a \in \{0, 1\}^n$ si para todo $x \in \{0, 1\}^n$ se cumple $f(x \oplus a) = f(x)$ y que es 2 a 1 si para todo $y \in \{0, 1\}^n$ se cumple que $f^{-1}(y)$ tiene cardinal 0 ó 2.

Problema 3 (*Problema de Simon*) Dada una función booleana $f : \{0, 1\}^n \rightarrow \{0, 1\}^n$, 2 a 1 y periódica, y su transformación unitaria asociada U_f , encontrar un algoritmo que calcule su periodo aplicando U_f el menor número posible de veces.

Es sencillo probar que clásicamente hay que evaluar f sobre la mitad más uno de los elementos del dominio, es decir sobre $2^{n-1} + 1$ elementos, para estar seguros de encontrar dos elementos x e y tales que $f(x) = f(y)$. A partir de x e y se calcula el periodo de la función $a = x \oplus y$. Si sólo buscamos una solución probabilística, con cota de error ϵ , habría que evaluar la función $O(2^{n/2})$ veces de forma que la probabilidad de encontrar x e y tales que $f(x) = f(y)$ sea mayor que $1 - \epsilon$, donde la constante que esconde la O depende de ϵ . Sin embargo, cuánticamente basta evaluar U_f unas pocas veces para obtener el periodo con cota de error ϵ . Por tanto, en la resolución probabilística de este problema el modelo cuántico de computación permite una ganancia exponencial respecto al modelo clásico.



Figure 15: Circuito para el problema de Simon

A continuación se muestra el estado del $(n+n)$ -qubit después de aplicar cada una de las transformaciones unitarias del circuito. El resultado final es un número entero y tal que $a \cdot y = 0$.

$$\begin{aligned} |0\rangle \otimes |0\rangle & \longrightarrow \frac{1}{\sqrt{2^n}} \sum_{0 \leq x < 2^n} (|x\rangle \otimes |0\rangle) \\ & \longrightarrow \frac{1}{\sqrt{2^n}} \sum_{0 \leq x < 2^n} (|x\rangle \otimes |f(x)\rangle) \\ & \longrightarrow \frac{1}{\sqrt{2}} (|x'\rangle + |x' \oplus a\rangle) \otimes |f(x')\rangle \\ & \longrightarrow \frac{1}{\sqrt{2^{n+1}}} \sum_{0 \leq y < 2^n} \left((-1)^{x' \cdot y} + (-1)^{(x' \oplus a) \cdot y} \right) |y\rangle \otimes |f(x')\rangle \\ & \longrightarrow y \quad \text{tal que} \quad a \cdot y = 0 \end{aligned}$$

La ecuación $a \cdot y = 0$ tiene n incógnitas (los dígitos binarios de a) y los coeficientes pertenecen a un cuerpo (cuerpo de enteros módulo 2). Debemos repetir el algoritmo hasta obtener un sistema lineal de ecuaciones homogéneo de rango $n - 1$. La solución no nula de las 2 que tiene este sistema es el periodo de la función f . El número de veces que hay que repetir el algoritmo para obtener un sistema lineal homogéneo de rango $n - 1$, con cota de error ϵ , es polinomial.

Ejercicios:

- Demostrar que el problema de Deutsch-Jozsa se puede resolver clásicamente con probabilidad de error inferior a ϵ haciendo $\lceil \log_2(2/\epsilon) \rceil$ evaluaciones de f .
Nota: $\lceil x \rceil$ representa al menor entero mayor o igual que x .
- Probar que para resolver clásicamente el problema de Simon con probabilidad de error inferior a ϵ hay que evaluar la función f al menos $2^{(n-1)/2} \sqrt{1-\epsilon}$ veces.
- Demostrar que la probabilidad de obtener un sistema homogéneo de rango $n - 1$ aplicando $n - 1$ veces el algoritmo de Simon es mayor que $\lim_{n \rightarrow \infty} \prod_{j=0}^{n-2} (1 - 2^{j-n+1})$.

5 Comentarios finales

El diseño de algoritmos cuánticos requiere una estrategia distinta de la que generalmente se utiliza para algoritmos clásicos. Éstos se plantean como una secuencia de cambios locales que realizan los cálculos deseados. Clásicamente estos cambios locales se corresponden con operaciones básicas del modelo de computación. Sin embargo en computación cuántica las operaciones básicas no son locales, tal como hemos visto al hablar de las puertas cuánticas. Entonces, para obtener algoritmos cuánticos eficientes es preferible olvidar el planteamiento clásico de los cambios locales.

References

- [1] Aharonov, D., “Quantum Computation”, Los Alamos Physics Preprint Archive <http://xxx.lanl.gov/abs/quant-ph/9812037>, (1998).
- [2] Barenco, A. y otros, “Elementary gates for quantum computation”, *Physical Review A*, **52**, 5 (1995), pp. 3457-3467.
- [3] Benioff, P., “Quantum mechanical Hamiltonian models of Turing machines”, *J. Stat. Phys.*, **29**, (1982), pp. 515-546.
- [4] Calderbank, A. R. y Shor, P. W., “Good quantum error-correcting codes exist”, Los Alamos Physics Preprint Archive <http://xxx.lanl.gov/abs/quant-ph/9512032>, (1996).
- [5] Deutsch, D., “Quantum theory, the Church-Turing principle and the universal quantum computer”, *Proc. of the Royal Society of London, Ser. A*, **A400**, (1985), pp. 97-117.
- [6] Dieks, D., “Communication by EPR Devices”, *Phys. Lett. A*, **92** (6), (1982), pp. 271-272.
- [7] Feynman R., “Simulating physics with computers”, *International Journal of Theoretical Physics*, **21**, 6-7 (1982), pp. 467-488.
- [8] Galindo, A. and Martín-Delgado, M. A., “Information and Computation: Classical and Quantum Aspects”, *Reviews of Modern Physics* (to appear), Los Alamos Physics Preprint Archive <http://xxx.lanl.gov/abs/quant-ph/0112105>, (2001).
- [9] Grover, L. K., “A fast quantum mechanical algorithm for database search”, Los Alamos Physics Preprint Archive <http://xxx.lanl.gov/abs/quant-ph/9605043>, (1996).

- [10] Nielsen, M. A. y Chuang, L. I., *Quantum Computation and Quantum Information*, Cambridge University Press, 2000.
- [11] Preskill, J., “Quantum Computation”, Lecture Notes <http://www.theory.caltech.edu/people/preskill/ph229>, (2000).
- [12] Rieffel, E. y Polak, W., “An Introduction to Quantum Computing for Non-Physicists”, Los Alamos Physics Preprint Archive <http://xxx.lanl.gov/abs/quant-ph/9809016>, (2000).
- [13] Rivest, R. L., Shamir, A. and Adleman, L. M., “A Method for Obtaining Digital Signatures and Public-Key Cryptosystems”, *Communications of the ACM*, **21** (2), (1978), pp. 120-126, [550].
- [14] Shor, P. W., “Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer”, Los Alamos Physics Preprint Archive <http://xxx.lanl.gov/abs/quant-ph/9508027>, (1994).
- [15] Steane, A., “Quantum computing”, Los Alamos Physics Preprint Archive <http://xxx.lanl.gov/abs/quant-ph/9708022>, (1997).
- [16] Wootters, W. K. and Zurek, W. H., “A Single quantum cannot be cloned”, *Nature*, **299**, (1982), pp. 802.

TRABAJOS PUBLICADOS

Nº	AUTOR(ES)	TITULO	FECHA
1 (I)	F. Gómez; S. Ramaswami; G. Toussaint	<i>On Removing Non-degeneracy Assumptions in Computational Geometry</i>	Abril 1996
2 (I)	P. Bose; F. Gómez; P. Ramos; G. Toussaint	<i>Drawing Nice Projections of Objects in Space</i>	Abril 1996
3 (I)	J. García	<i>A Counter-Example to a Theorem of Sendov</i>	Abril 1996
4 (I)	J. García; P.A. Ramos	<i>Circularity of a set of points</i>	Abril 1996
5 (I)	F. Gómez; F. Hurtado, G. Toussaint	<i>Proyecciones de Calidad y Reconstrucción de Conjuntos</i>	Octubre 1996
6 (D)	O.F. Soto; E. Osejo	<i>Una numeración biyectiva de los racionales</i>	Octubre 1996
7 (I)	M. Villén; J. Villén	<i>RESTART: An efficient and general method for fast simulation of rare events.</i>	Julio 1997
8 (I)	F. García	<i>Convergence Theorems for Topological Group Valued Measures on Effect Algebras.</i>	Septiembre 1997
9 (D)	A. García	<i>Informática y Matemáticas (Historia de un matrimonio de conveniencia).</i>	Diciembre 1997
10 (D)	J. García	<i>Algoritmos eficientes de enumeración I.</i>	Junio 1998
11 (I)	J.M. Díaz-Báñez; F. Gómez; F. Hurtado	<i>Some Problems on Approximation of Set of Points by Polygonal Curves.</i>	Abril 1999
12 (I)	A. García	<i>Maple and the z-transform.</i>	Noviembre 1999
13 (D)	F. Alonso; A. García; F. García; S. Hoya; G. Rodríguez; A. de la Villa	<i>Some unexpected results using Computer Algebra Systems.</i>	Julio 2000
14 (D)	A. Franco; P. Franco; A. García; F. García; F.J. González; S. Hoya; G. Rodríguez; A. de la Villa	<i>Learning Calculus of several variables with new technologies.</i>	Julio 2000
15 (I)	J. García	<i>Demostración en Lógica Proposicional</i>	Marzo 2001
16 (D)	J. García	<i>Analizador léxico del lenguaje Ahmes</i>	Junio 2001
17 (I)	M. Villén; J. Villén	<i>Analysis of RESTART Simulation: Theoretical Basis and Sensitivity Study</i>	Noviembre 2001
18 (D)	A. García López	<i>Algoritmo de búsqueda de Grover</i>	Junio 2003
19 (D)	Grupo de Computación Cuántica	<i>Introducción al modelo cuántico de computación</i>	Junio 2003