Universidad de Oviedo

Facultad de Ciencias

Máster Universitario en Modelización e Investigación Matemática, Estadística y Computación

Trabajo de fin de máster

Criptografía y códigos correctores cuánticos

Autor: Ángel José Riesgo Martínez

Tutor: Ignacio Fernández Rúa

Julio de 2015

A mi hermana, a mi madre y a la memoria de mi padre

Agradecimientos

La preparación de este trabajo de fin de máster ha ocupado una gran parte de mi tiempo durante este último año, lo cual, inevitablemente, ha acabado afectando también a mi vida personal y laboral. Por ello, tengo que agradecer el apoyo de las personas que han estado cerca de mí durante este tiempo; en particular, doy las gracias a mi familia, mi hermana y mi madre, y a mis amigos más próximos, que en todo momento me han animado a perseverar en el esfuerzo hasta completar los estudios de este máster universitario.

Para la realización del trabajo en sí, he tenido la inmensa fortuna de contar con la supervisión de Iñaki, que ha mostrado en todo momento una enorme paciencia al esperar por mis borradores parciales, que yo le iba enviando a mi ritmo, más lento del que habría deseado, casi siempre con retraso, y que él revisaba y corregía prácticamente de inmediato, incluso en momentos en que se encontraba de viaje o vacaciones. Sus correcciones, sugerencias y comentarios han contribuido a elevar la calidad y el nivel matemático del trabajo, muy pobre en mis versiones iniciales, y me han ofrecido una oportunidad única para adentrarme, aun mínimamente, en un campo tan fascinante como es el de la computación cuántica.

Criptografía y códigos correctores cuánticos

Máster en Modelización e Investigación Matemática, Estadística y Computación

Ángel José Riesgo Martínez

20 de julio de 2015

Resumen

Este documento se ha presentado en julio de 2015 como trabajo final para el máster universitario en Modelización e Investigación Matemática, Estadística y Computación de la Universidad de Oviedo. El trabajo comienza con una introducción a los principios matemáticos de la computación cuántica y continúa con una descripción de los principales algoritmos cuánticos. A continuación, se abordan las repercusiones de estos algoritmos sobre la criptografía actual y las dificultades tecnológicas que impiden hoy en día el desarrollo de ordenadores cuánticos. La segunda mitad del trabajo aborda uno de los campos de estudio más relevantes desde un punto de visto matemático de estas dificultades prácticas, que es la necesidad de mecanismos de corrección de errores, mucho más críticos que en la computación basada en la física clásica. Finalmente, se concluye que pese a los progresos de las últimas dos décadas son aún muchas las dificultades tanto prácticas como teóricas para desarrollar ordenadores cuánticos.

Abstract

This document has been presented in July 2015 as a final dissertation for the University of Oviedo MSc degree course in Mathematical Modelling and Research, Statistics and Computing. The work begins with an introduction to the mathematical principles behind quantum computing and continues with a description of the main quantum algorithms. After that, we analyse the impact of these algorithms on current cryptography and the technological difficulties that hinder the development of quantum computers at present. The second half of the document tackles one of the most relevant fields of study, from a mathematical standpoint, regarding these practical difficulties, which is the need for error correction techniques. Such techniques are much more critical than in the computation based on classical physics. Finally, we conclude that despite the progress made over the last two decades, there are still too many difficulties, both practical and theoretical, that prevent the development of quantum computers.

Índice

1 Aspectos generales de la computación cuántica

1.1 Los postulados de la mecánica cuántica

La teoría de la computación cuántica se basa en la representación de la información mediante bits cuánticos o *qubits*, cuyo estado físico no es puramente binario, restringido a dos valores posibles, sino que admite la existencia de estados superpuestos que combinan los dos valores. Esa posibilidad de la superposición de estados, que se deriva de las leyes cuánticas aplicables a los sistemas físicos en escalas atómicas y subatómicas, da lugar a la posibilidad de describir algoritmos basados en puertas lógicas diferentes de las utilizadas en la computación clásica, en la que la unidad de información, el bit, solamente admite dos estados, típicamente representados como "0" y "1". En esta primera sección expondremos las ideas más básicas de la mecánica cuántica que son necesarias para entender las características tan peculiares de los *qubits*.

La mecánica cuántica, que describe correctamente el mundo físico en escalas atómicas y subatómicas, se diferencia de una manera fundamental de la mecánica clásica con la que se modela el mundo macroscópico. Estas diferencias fundamentales afectan a la propia naturaleza de los sistemas físicos estudiados. Mientras que en la mecánica clásica el conocimiento completo de un sistema formado por varios objetos equivale a conocer las trayectorias de estos objetos, en la mecánica cuántica desaparece el concepto de "trayectoria" y no es posible determinar la posición y la velocidad de un cuerpo en todo instante, sino solamente la probabilidad de que, en el instante en que se toma una medida, el cuerpo esté en una cierta posición o, en general, que una de sus propiedades físicas, como podrían ser la posición, el momento lineal o angular, la energía, el espín, etc., revelen un determinado valor. En este sentido, dos conceptos claves de la descripción cuántica de la realidad son el papel de la medición como algo implícito en el modelo físico y la naturaleza probabilística de los valores medidos. Mientras que en la mecánica clásica la medición del sistema es algo accesorio a la teoría (se asume que siempre existen valores exactos de las magnitudes físicas independientemente de que haya o no una observación o medición), en la mecánica cuántica las "trayectorias" no ocurren en un espacio euclídeo de posiciones, sino en un espacio de probabilidades en el que lo que evoluciona con el tiempo es la probabilidad de que la observación, en caso de producirse, arroje un determinado valor. Además, la observación altera estas trayectorias probabilísticas pues, evidentemente, en el mismo instante de la medición se pasa a conocer con exactitud, probabilidad 1, el valor medido.

La reinterpretación del propio modelo de "sistema físico" introducida por la mecánica cuántica fue uno de los debates intelectuales más intensos de la filosofía de la ciencia en el siglo pasado. Pero más allá del debate interpretativo, el formalismo matemático subyacente es muy simple y elegante. El primer intento de dar una base matemática más sólida a estas nuevas ideas corresponde a Paul A. M. Dirac, que en su obra *The Principles of Quantum Mechanics* [10] de 1930 introdujo el formalismo

y la notación utilizados actualmente del espacio de estados como un espacio vectorial en el que las magnitudes físicas observables corresponden a endomorfismos. Este formalismo fue descrito de forma más rigurosa por John von Neumann en su obra *Mathematische Grundlagen der Quantenmechanik* [20], publicada en 1932, en la que identificaba el espacio vectorial de estados como un espacio de Hilbert en el que desempeñaban un papel fundamental los endomorfismos unitarios. Aún hoy en día, no hay un consenso en la literatura sobre los principios básicos que se necesitan para formalizar matemáticamente la mecánica cuántica y los axiomas o postulados varían según las fuentes dependiendo del rigor formal. Siguiendo la descripción de Umesh Vazirani [27], podemos identificar tres axiomas o postulados básicos que nos servirán de punto de partida para poder trabajar matemáticamente con sistemas cuánticos. Enumeremos a continuación estos tres axiomas.

1. **El principio de superposición**. Dado un sistema físico que admite varios estados observables diferentes (en el sentido de que la medición de una determinada magnitud arroja valores diferentes), estos estados corresponden a vectores ortonormales de un espacio de Hilbert sobre el cuerpo $\mathbb{C}$ de los números complejos. El estado del sistema en un instante temporal t viene dado por un elemento cualquiera del espacio de Hilbert de norma 1. En otras palabras, el estado del sistema es en general una combinación lineal o superposición de los estados observables (base ortonormal del espacio vectorial) en que los coeficientes de cada componente están sujetos a una condición de normalización que hace que sean vectores en la hiperesfera de radio 1 del espacio de Hilbert.

2. **El principio de observación**. Cuando se toma una medición (u observación) de la magnitud física, el estado del sistema se altera y pasa a ser uno de los estados de la base. A este fenómeno tan peculiar de la mecánica cuántica, en que la mera observación modifica el estado del sistema, se le denomina en la terminología habitual de los físicos el "colapso del estado". Se dice pues que el estado de superposición en el que había varias componentes no nulas colapsa (en castellano a veces "decae") sobre el estado observable, que coincide con uno de los vectores de la base del espacio de Hilbert. La interpretación de este colapso del estado es probabilística: si, por ejemplo, la energía de una partícula puede observarse en dos valores diferentes E_0 y E_1, estos valores tendrán asociados dos vectores ortonormales $\mathbf{v_{E_0}}$ y $\mathbf{v_{E_1}}$, respectivamente, en el espacio de Hilbert. Si el estado actual es $\mathbf{v} = \sqrt{0,1}\mathbf{v_{E_0}} + \sqrt{0,9}\mathbf{v_{E_1}}$, entonces la medición de la energía provocará que el estado pase a ser $\mathbf{v_{E_0}}$ con probabilidad 0,1 o $\mathbf{v_{E_1}}$ con probabilidad 0,9 y la energía medida sería E_0 y E_1 en cada caso. Nótese que si el estado no observado fuera exactamente $\mathbf{v_{E_0}}$, entonces sabemos con certeza (probabilidad 1) que la observación arrojará el valor E_0. Las magnitudes físicas como la energía pueden considerarse endomorfismos hermíticos en el espacio de Hilbert, cuyos estados observables constituyen los autovectores del espacio y cuyos valores medidos son los autovalores correspondientes. La condición de que estos endomorfismos sean hermíticos significa que una vez elegida una base, la matriz coordenada del endomorfismo es una matriz que es igual a su transpuesta conjugada; esto garantiza que los autovalores sean reales. Esta relación entre magnitudes observables y endomorfismos hermíticos puede tomarse como parte de la axiomática o bien deducirse como consecuencia de la interpretación probabilística del colapso

del estado.

3. **La evolución unitaria**. Mientras no se produzca una operación de observación como las descritas por el segundo postulado, el sistema físico cuántico puede evolucionar en el tiempo de acuerdo con transformaciones lineales y unitarias en el espacio de Hilbert. Los endomorfismos unitarios en un espacio de Hilbert complejo son aquellos endomorfismos que tienen como inverso el endomorfismo adjunto. En el ejemplo con dos valores posibles para la energía E_0 y E_1, tendríamos que el sistema está en un estado que es igual a $\mathbf{v_{E_0}}$ o a $\mathbf{v_{E_1}}$ una vez que se lo observa, pero con el transcurso del tiempo tras la observación, el estado evoluciona hacia una superposición $\alpha\mathbf{v_{E_0}} + \beta\mathbf{v_{E_1}}$ tal que $\alpha^2 + \beta^2 = 1$ y los coeficientes (α, β) se obtienen a partir del estado inicial $(1, 0)$ o $(0, 1)$ mediante un endomorfismo cuya matriz coordenada U sería una matriz compleja que respete la condición de ser unitaria $UU^\dagger = U^\dagger U = I$, en donde $U^\dagger$ es la matriz resultado de reemplazar en U cada entrada por su compleja conjugada y transponer.

En aquellos casos en que las magnitudes físicas pueden adoptar un continuo de valores la teoría se complica, ya que el espacio de Hilbert de estados pasa a tener dimensión infinita y la probabilidad discreta deviene una densidad de probabilidad continua. Esta situación da lugar al formalismo basado en funciones de onda, que equivale a la descripción de la evolución cuántica debida a Erwin Schrödinger [25]. En nuestro estudio pasaremos por alto estos casos y nos limitaremos a la situación en que el espacio de Hilbert es de dimensión finita. Esto no supone ningún tipo de simplificación, ya que en la mecánica cuántica surgen de manera natural las magnitudes que solamente pueden adoptar un conjunto discreto de valores (el propio nombre de "mecánica cuántica" se deriva de esa cuantización esencial de los sistemas atómicos y subatómicos). Es más, de hecho solamente nos interesan los sistemas que tienen exactamente dos estados observables, que podemos identificar con los valores 0 y 1 de un bit de información (y los sistemas compuestos por este tipo básico de sistema, como veremos). La naturaleza física de ese par de estados es irrelevante. Podrían ser dos niveles de energía de un átomo, dos estados de superconductividad o los dos posibles espines de una partícula. Lo que nos interesa es que tales sistemas con dos estados observables son viables experimentalmente y conducen a la idea del *qubit*, la generalización al mundo cuántico del concepto clásico de bit como unidad de información.

1.2 El equivalente cuántico del bit: el *qubit*

El concepto de *qubit* se deriva de las características que acabamos de describir para los sistemas cuánticos. Mientras que en la física clásica una magnitud de tipo binario con dos valores posibles 0 y 1 evolucionará adoptando exclusivamente uno de esos dos valores en cualquier instante de tiempo t, en el caso de un sistema cuántico la existencia de dos únicos valores posibles 0 y 1 para una magnitud afecta al momento en que se mide dicha magnitud, pero no a su evolución temporal, ya que mientras el sistema no es sometido a medición, el principio de superposición establece que su

estado vendrá dado por una combinación lineal de los estados 0 y 1, autovectores en el espacio de Hilbert que caracteriza al sistema. Y la manera en que evoluciona el sistema en el tiempo vendrá dada por un endomorfismo unitario del espacio de acuerdo con el postulado de evolución unitaria.

1.3 La notación *bra-ket* de Dirac

Para trabajar con los estados de estos espacios de Hilbert resulta conveniente adoptar la notación *bra-ket* debida a Paul Dirac. En esta notación, los vectores base del espacio de un *qubit* se designan como $|0\rangle$ y $|1\rangle$. Si combinamos linealmente los dos vectores base para obtener otro estado con coeficientes λ_0 y λ_1 (tales que cumplan la condición de normalización $|\lambda_0|^2 + |\lambda_1|^2 = 1$), tendremos un vector de estado $|\psi\rangle = \lambda_0|0\rangle + \lambda_1|1\rangle$. A los vectores designados de esta manera Dirac los llamó *kets* y es un tipo de notación que ha tenido éxito porque muestra explícitamente el estado físico que representa el vector base. Si en lugar de considerar los estados como 0 y 1 tuviéramos dos estados de energía E_0 y E_1 llamaríamos a los vectores base $|E_0\rangle$ y $|E_1\rangle$. Si se tratara de un estado de espín o de un estado de polarización con dos orientaciones posibles podemos utilizar las notaciones $|\uparrow\rangle$ y $|\downarrow\rangle$. De esta manera, el estado al que corresponde cada componente se muestra de manera explícita, en contraste con la notación vectorial convencional de pares de componentes. Además, esta notación de los *kets* se extiende a las formas lineales del espacio mediante los llamados *bras*: $\langle 0|$, $\langle 1|$, $\langle E_0|$, $\langle E_1|$, $\langle\uparrow|$ y $\langle\downarrow|$, que serían elementos del espacio dual. Aquí se hace evidente el origen del nombre de la notación en la palabra inglesa *bracket* (paréntesis): cuando aplicamos una forma lineal a un vector, tendremos un *bra* seguido de un *ket*; por ejemplo, $\langle\phi|\psi\rangle$, con el que obtenemos un número. Los paréntesis cerrados corresponden así a números, el producto interno del espacio de Hilbert, mientras que las expresiones que solamente constan de la parte izquierda o de la parte derecha corresponderán a formas lineales o vectores, respectivamente. En esta notación de Dirac, los endomorfismos se representan habitualmente mediante letras mayúsculas como P, con lo que $P|\psi\rangle$ sería el vector transformado por P, mientras que $\langle\phi|S$ sería una forma lineal $\langle\phi|$ transformada por un endomorfismo S del espacio dual. En el caso de que el endomorfismo sea autoadjunto o hermítico, $P = P^\dagger$, se tiene que $(\langle\phi|P)|\psi\rangle = \langle\phi|(P^\dagger|\psi\rangle) = \langle\phi|(P|\psi\rangle)$ y se puede escribir, sin ambigüedad, $\langle\phi|P|\psi\rangle$. En tal caso, P se puede considerar también como una forma bilineal en el espacio de Hilbert. Cuando el endomorfismo hermítico P corresponde a una magnitud física observable, el valor numérico real $\langle\psi|P|\psi\rangle$ tiene una interpretación probabilística como valor esperado de la medición de la magnitud física P para el sistema en el estado descrito por el *ket* $|\psi\rangle$.

1.4 Los sistemas compuestos

Otro concepto importante para entender los fundamentos de la computación cuántica es el de la descripción de sistemas compuestos. Dados dos sistemas físicos, pode-

mos tratarlos como uno solo mediante la definición apropiada de un nuevo espacio de estados que incorpore las mediciones en los dos sistemas básicos. En el caso que nos interesa, dados dos *qubits* podemos considerar las observaciones conjuntas en que podemos obtener cuatro valores $\{(0,0),(0,1),(1,0),(1,1)\}$. Uno de los resultados básicos de la mecánica cuántica, que a veces se considera parte de los postulados, define este espacio de estados compuesto como el producto tensorial de los espacios de estados simples. Esto quiere decir que una base ortonormal del espacio de Hilbert para el sistema de dos *qubits* sería $\{|0\rangle\otimes|0\rangle, |0\rangle\otimes|1\rangle, |1\rangle\otimes|0\rangle, |1\rangle\otimes|1\rangle\}$. Simplificando la notación, escribiremos estos estados básicos simplemente como $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$. La composición puede extenderse a sistemas con un número natural arbitrario n de *qubits*, cuyo espacio de Hilbert tendría 2^n elementos $\{|0\ldots0\rangle, \ldots, |1\ldots1\rangle\}$. En los sistemas de n *qubits* se definen análogamente los conceptos de endomorfismos hermíticos y unitarios, que pueden construirse también como productos tensoriales de los endomorfismos definidos en los subsistemas de menor número de *qubits*. Los procesos de medición pueden afectar a todo el sistema compuesto o bien parcialmente a un subsistema. En este último caso, tras la medición parcial, el estado decae a un vector en el subespacio de Hilbert complemento ortogonal del subespacio que ha sido sometido a medición.

En esta sección, hemos introducido las notaciones y los conceptos que se necesitan para la descripción de la computación cuántica de una manera intuitiva. En la sección siguiente, definiremos de una manera más formal y rigurosa los conceptos básicos.

2 El formalismo matemático

2.1 El espacio de estados de un sistema cuántico

Tras la introducción informal de la sección anterior, en esta sección vamos a presentar de manera formal los conceptos que necesitaremos en el resto del trabajo. Para ello, asumimos como concepto primitivo que existen sistemas físicos a los que llamaremos **sistemas cuánticos** cuyas propiedades pueden describirse mediante la teoría de la mecánica cuántica y empezaremos definiendo el tipo de espacio matemático sobre el que se construye esta teoría física.

Definición 2.1. Se llama **espacio de estados** de un sistema cuántico a un espacio vectorial H de dimensión finita d sobre el cuerpo $\mathbb{C}$ de los números complejos y que está dotado de un producto interno que le da estructura de espacio de Hilbert.

Utilizando la notación de Dirac introducida en la sección anterior, a los vectores de un espacio de estados los representaremos mediante la notación *ket*; por ejemplo, $|\psi\rangle$. Al producto interno entre dos vectores $|\phi\rangle$ y $|\psi\rangle$ lo representaremos como $\langle\phi|\psi\rangle$. Con esta notación, el producto interno que da a un espacio vectorial estructura de espacio de Hilbert ha de cumplir, para cualesquiera vectores del espacio y para cualesquiera números complejos, las siguientes propiedades [22, p. 181]:

$$1) \qquad \langle\phi|\psi\rangle \geqslant 0 \qquad \text{y} \qquad \langle\psi|\psi\rangle = 0 \iff |\psi\rangle = 0$$

$$2) \qquad \langle\phi|\psi\rangle = \overline{\langle\psi|\phi\rangle} \tag{2.1}$$

$$3) \qquad \langle\lambda_1\phi_1 + \lambda_2\phi_2|\psi\rangle = \lambda_1\langle\phi_1|\psi\rangle + \lambda_2\langle\phi_2|\psi\rangle$$

La segunda condición tiene como consecuencia inmediata que el producto interno de un vector del espacio de Hilbert consigo mismo es siempre un número real.

Aquí debemos hacer dos observaciones sobre la notación *bra-ket*. Por un lado, en la última de las expresiones anteriores hay un frecuente abuso de notación, por el que se acepta escribir $|\lambda_1\phi_1 + \lambda_2\phi_2\rangle$ en lugar de $\lambda_1|\phi_1\rangle + \lambda_2|\phi_2\rangle$. Por otra parte, la notación $\langle\phi|\psi\rangle$ que utilizamos para el producto interno de dos vectores $|\phi\rangle$ y $|\psi\rangle$ puede interpretarse también como la aplicación de una forma lineal $\langle\phi|$ sobre un vector $|\psi\rangle$. Ambas interpretaciones son en cualquier caso equivalentes en un espacio vectorial dotado de producto interno, debido a que en tales espacios existe un isomorfismo canónico entre el espacio vectorial y su dual. Y este isomorfismo canónico se define precisamente mediante el producto interno. Por ello, la notación $\langle\phi|\psi\rangle$ no conlleva ningún tipo de ambigüedad.

Al disponer de un producto interno, se puede definir de la manera habitual una norma en el espacio de estados:

Definición 2.2. Dado un espacio de estados H y un elemento cualquiera $|\psi\rangle \in H$, se llama **norma**, **módulo** o **longitud** de $|\psi\rangle$ al número real $||\psi|| = \sqrt{\langle\psi|\psi\rangle}$.

Gracias a esta definición de norma, que da al espacio de estados estructura de espacio normado, se puede definir el concepto de base ortonormal:

Definición 2.3. Dado un espacio de estados H de dimensión d, se llama **sistema ortonormal de vectores** a un conjunto de k vectores $\{|\psi_i\rangle\}_{i=1,\ldots,k}$ con $0 < k \leqslant d$ tales que $\langle\psi_i|\psi_j\rangle = 0$ para $i \neq j$ y $\langle\psi_i|\psi_i\rangle = 1$. Cuando el numero de vectores k es igual a la dimensión d del espacio de Hilbert, el sistema ortonormal es una base del espacio y se lo denomina **base ortonormal**.

Nótese que en espacios de Hilbert de dimensión infinita se hace una distinción entre las "bases de Hamel" de la estructura vectorial y las "bases de Hilbert" de la estructura debida al producto interno, pero esta distinción es innecesaria en espacios de dimensión finita, que son los únicos que nos interesan en el contexto de la computación cuántica.

Proposición 2.1. *Sea un espacio de estados H de dimensión d con una base ortonormal formada por vectores $\{|e_i\rangle\}_{i=1,\ldots,d}$. Dados dos vectores cualesquiera $|\phi\rangle$ y $|\psi\rangle$, cuyas expresiones en función de la base son $|\phi\rangle = \sum_{i=1}^{d} a_i|e_i\rangle$ y $|\psi\rangle = \sum_{i=1}^{d} b_i|e_i\rangle$ ($a_i, b_i \in \mathbb{C}$; $i = 1, \ldots, d$), entonces su producto interno es igual a:*

$$\langle \phi | \psi \rangle = \sum_{i=1}^{d} \overline{a_i} b_i \tag{2.2}$$

Demostración. Basta con desarrollar $\langle \phi | \psi \rangle$ como $(\sum_{i=1}^{d} a_i \langle e_i |)(\sum_{j=1}^{d} b_j | e_j \rangle)$ y aplicar las condiciones (2.1) hasta que se cancelen los vectores $|e_i\rangle$ mediante las condiciones de ortonormalidad. □

De esta manera, tenemos ya las propiedades fundamentales del espacio matemático que nos da la descripción de los estados observables de un sistema físico. En lo sucesivo, emplearemos indistintamente las expresiones "espacio de estados" y "espacio de Hilbert", asumiendo siempre que se trata de un espacio de Hilbert de dimensión finita sobre los números complejos.

Pero, en realidad, no todos los vectores del espacio de estados tienen una interpretación física distinguible. La norma de los vectores es irrelevante, por lo que se suelen considerar únicamente vectores con norma 1 (a los que a veces se llama **vectores normalizados** o **vectores unitarios**). Tampoco cambia la interpretación física el producto de un vector por un número complejo, por lo que cualquier factor que multiplique a un vector puede considerarse irrelevante en este sentido. Esta idea podemos representarla matemáticamente mediante una relación de equivalencia:

Definición 2.4. Dado un espacio de estados H, podemos definir una relación de equivalencia $\sim$ en el conjunto de los vectores de H tal que dos vectores son equivalentes cuando uno es igual al otro multiplicado por un número complejo; es decir, dados $|\phi\rangle, |\psi\rangle \in H$, decimos que $|\phi\rangle \sim |\psi\rangle$ si existe $\lambda \in \mathbb{C}$ tal que $|\psi\rangle = \lambda |\phi\rangle$.

Definición 2.5. Dado un espacio de estados H y la relación $\sim$ de la definición (2.4) anterior, al conjunto cociente $H/\sim$ se lo denomina **espacio de estados físicos** asociado a H. A cada uno de sus elementos se lo denomina **estado físico puro**.

La definición anterior recoge la idea de que los estados físicamente diferenciables son aquellos que corresponden a vectores unitarios del espacio de Hilbert ignorando un factor complejo arbitrario. De este modo, dado un vector unitario $|\psi\rangle$, el vector $re^{i\theta}|\psi\rangle$ $(r, \theta \in \mathbb{R})$ representa el mismo estado observable; el factor r se puede eliminar normalizando el vector y el factor complejo $e^{i\theta}$, al que se suele llamar "fase", carece de relevancia física (aunque puede ser importante en los cálculos). A estos estados los llamaremos "puros" para distinguirlos de un concepto más amplio de estado, los "estados mixtos", que definiremos en una sección posterior. Aunque esta distinción entre el espacio de estados físicos y el espacio de Hilbert al que hemos denominado simplemente espacio de estados resulta útil, no se suele tener en cuenta y en el resto de este trabajo llamaremos por lo general "estados" a los vectores del espacio de Hilbert, asumiendo siempre implícitamente que solamente tienen interés los vectores unitarios e ignorando, en la mayoría de los casos, los factores de fase.

Definiremos a continuación formalmente los dos tipos de endomorfismos en espa-

cios de Hilbert complejos que tienen relevancia en el marco de la teoría cuántica.

Definición 2.6. Dado un espacio de estados H y un endomorfismo $T : H \to H$ se llama **endomorfismo adjunto** a otro endomorfismo $T^\dagger$ tal que para todo vector $|\psi\rangle \in H$ se cumple $\langle\psi|(T|\phi\rangle) = (\langle\psi|T^\dagger)|\phi\rangle$.

Se puede demostrar que para todo endomorfismo del espacio de estados, el endomorfismo adjunto existe siempre y es único [22, p. 201]. En notación matricial fijada una base ortonormal, la matriz coordenada de $T^\dagger$ corresponde a la matriz transpuesta de T con las entradas reemplazadas por las correspondientes complejas conjugadas.

Definición 2.7. Dado un espacio de estados H y un endomorfismo $T : H \to H$, se dice que T es **hermítico** o **autoadjunto** si cumple $T = T^\dagger$.

Estos endomorfismos hermíticos desempeñan un papel muy importante en la teoría cuántica porque, tal como adelantamos en la sección introductoria, representan magnitudes físicas observables. Cuando el vector del espacio de Hilbert puede interpretarse como un estado físico del sistema y el endomorfismo hermítico T representa una magnitud observable, podemos definir un valor que coincide con el concepto estadístico de valor esperado de una medición:

Definición 2.8. Dado un vector unitario $|\psi\rangle$ de un espacio de estados H y un endomorfismo hermítico $T : H \to H$, se llama **valor esperado de la magnitud observable T en el estado** $|\psi\rangle$ al valor $\langle\psi|T|\psi\rangle$.

Para que esta definición corresponda realmente al resultado de una medición, $\langle\psi|T|\psi\rangle$ no debería tener parte imaginaria. La siguiente proposición, una propiedad de entre muchas análogas enumeradas en el libro de álgebra lineal de Steven Roman [22, p. 209], garantiza esto.

Proposición 2.2. *Dado un espacio de estados H, en el que se define un endomorfismo hermítico $T : H \to H$, el producto interno $\langle\psi|T|\psi\rangle$ es real para todo $|\psi\rangle \in H$.*

Demostración. Al ser T hermítico, para todo vector $|\psi\rangle$ se cumple:

$$\langle\psi|T|\psi\rangle = \overline{\langle\psi|T^\dagger|\psi\rangle} = \overline{\langle\psi|T|\psi\rangle} \tag{2.3}$$

□

En el marco de la teoría cuántica, se utiliza también un tipo de transformación muy peculiar que corresponde al proceso físico de la observación. Esta es la definición que plantea más dificultades formales.

Como paso previo a la definición formal del proceso de observación, necesitamos dos resultados importantes, que enumeramos a continuación y que también se encuentran en el libro de S. Roman.

Proposición 2.3. *En un espacio de estados H, los autovalores de un endomorfismo hermítico $T : H \to H$ son reales [22, p. 209].*

Demostración. Sea λ un autovalor de T. Entonces existe un autovector $|\psi\rangle$ tal que $T|\psi\rangle = \lambda|\psi\rangle$. Por el resultado anterior (2.2), el producto interno $\langle\psi|T|\psi\rangle$ es un número real y se tiene:

$$\langle\psi|T|\psi\rangle = \lambda\langle\psi|\psi\rangle \tag{2.4}$$

Luego λ es real.

□

Proposición 2.4. *Sea un espacio de estados H en el que está definido un endomorfismo hermítico T. Entonces existe una base ortonormal de H formada por autovectores de T.*

Demostración. Se trata de un caso particular del Teorema de estructura para operadores normales (los endomorfismos hermíticos son un caso particular de los normales), cuya demostración puede encontrarse en el libro de S. Roman [22, p. 216]. □

Teniendo en cuenta los resultados anteriores, podemos definir ya la transformación de observación.

Definición 2.9. Dado un vector unitario $|\psi\rangle$ de un espacio de estados H de dimensión d y un endomorfismo hermítico $T : H \to H$, llamamos **observación de la magnitud observable T en el estado** $|\psi\rangle$ a una transformación que hace corresponder a $|\psi\rangle$ un nuevo estado que puede ser uno cualquiera de los autovectores de T. El autovector concreto resultado de la transformación depende de un comportamiento probabilístico: si el estado $|\psi\rangle$ se descompone en función de una base de autovectores $\{e_i\}_{i=1,\dots,d}$ con coordenadas $\{a_i\}_{i=1,\dots,d}$ ($|\psi\rangle = \sum_{i=1}^{d} a_i|e_i\rangle$), entonces la probabilidad de que la observación transforme a $|\psi\rangle$ en el autovector $|e_i\rangle$ es igual a $|a_i|^2$. En tal caso, al autovalor λ_i correspondiente al autovector $|e_i\rangle$ se lo denomina **resultado de la observación**.

Estas transformaciones de observación pueden definirse también sobre subespacios ortogonales, con un comportamiento probabilístico análogo. En tales casos, hablaremos de **observaciones parciales**.

La terminología escogida en la definición anterior alude evidentemente a la interpretación física de este tipo de transformación, asociada a los procesos de medición. Es importante hacer notar que este concepto de observación o medición no tiene por qué ser debido a la acción de un ser humano o consciente, sino que tales observaciones surgen de forma espontánea en los fenómenos naturales. A la aparición de transformaciones de observación se la denomina "decoherencia".

Además de los endomorfismos hermíticos, hay otro tipo de aplicaciones lineales muy importantes en los espacios de Hilbert complejos, que vamos a definir a continuación:

Definición 2.10. Dado un espacio de Hilbert complejo H y un endomorfismo $U : H \to H$, se dice que U es **unitario** si cumple $UU^\dagger = U^\dagger U = I$.

El interés de este tipo de endomorfismos se debe a que mantienen constante el producto interno, por lo que transforman una base ortonormal del espacio en otra y, en ese sentido, son análogos a los endomorfismos ortogonales en espacios vectoriales reales. Como vimos en la sección introductoria, la evolución de un sistema cuántico en el tiempo cuando no hay observación viene dada por un endomorfismo unitario. Para evitar en lo sucesivo alusiones vagas a un concepto indefinido de "tiempo", nos detendremos primero a establecer una definición formal del concepto de evolución temporal.

2.2 La evolución temporal de un sistema cuántico

La idea física de la evolución temporal de un sistema puede formalizarse matemáticamente como una aplicación de un conjunto de valores temporales en el espacio de estados, de modo que a cada instante de tiempo se le asigne un estado del sistema.

Definición 2.11. Dado un espacio de estados H, llamamos **evolución temporal** del espacio para un estado inicial dado $|\psi_0\rangle \in H$ a una aplicación $\psi : T \to H$, en donde T es un conjunto en principio arbitrario a cuyos elementos denominamos **instantes de tiempo** y que contiene al menos un elemento t_0 para el que se cumple $\psi(t_0) = |\psi_0\rangle$. De manera axiomática, asumimos que dados dos instantes de tiempo cualesquiera $t_1, t_2 \in T$, existe un endomorfismo unitario U sobre H tal que $\psi(t_2) = U\psi(t_1)$.

Aunque en física es habitual identificar el tiempo con la recta real, lo que correspondería a $T = \mathbb{R}$ en la anterior definición (2.11), en el estudio de la computación cuántica se puede tratar el tiempo como si fuera un conjunto discreto de etapas, obviamente finitas. Por consiguiente, puede tomarse un conjunto T finito. Con esta definición de conjunto T, la evolución temporal se convierte en una aplicación de un subconjunto finito de los enteros sobre el espacio de Hilbert H, lo que equivale al concepto matemático de sucesión finita.

Esta definición de evolución temporal como sucesión finita de n vectores de un espacio de Hilbert implica la existencia de una relación de orden estricto total. Así, dada una evolución temporal de estados $\{|\psi_i\rangle\}(1 \leqslant i \leqslant n)$, para cada estado $|\psi_i\rangle$ con $i < n$ existirá un endomorfismo unitario (que puede ser la identidad) U_i tal que el estado siguiente $|\psi_{i+1}\rangle$ cumple $|\psi_{i+1}\rangle = U_i|\psi_i\rangle$. Y dados dos estados $|\psi_i\rangle$ y $|\psi_j\rangle$ con $j > i$, se tendrá que $|\psi_j\rangle = U_{i\to j}|\psi_i\rangle$, donde $U_{i\to j}$ es el endomorfismo unitario $U_{i\to j} = \prod_{k=j-1}^{i} U_k$.

2.3 Los sistemas de información cuántica: los *qubits*

Hasta aquí hemos visto algunos conceptos básicos aplicables a cualquier sistema físico descrito mediante la teoría cuántica. Pero en el ámbito de este trabajo, no nos interesan los sistemas físicos generales, sino simplemente el sistema idealizado en el que hay dos valores observables diferentes. En este caso, podemos definir como un espacio de estados la unidad básica de información cuántica, el *qubit*.

Definición 2.12. Se llama **sistema de un *qubit*** a un espacio de estados de dimensión 2. Llamaremos ***qubit*** a un vector cualquiera de este espacio. Una vez fijada una base ortonormal del sistema de un *qubit*, sus dos vectores se representan habitualmente como $\{|0\rangle, |1\rangle\}$.

Como vimos en la sección introductoria, los sistemas compuestos se obtienen a partir de productos tensoriales. Esto nos permite dar una definición formal para los sistema de varios *qubits*:

Definición 2.13. Se llama **sistema de n *qubits*** al producto tensorial de n *qubits*. Si el *qubit* i-ésimo tiene base $\{|0\rangle_i, |1\rangle_i\}$, entonces una base de un sistema de n *qubits* consta de 2^n elementos $\{\left|0 \overset{n)}{\ldots} 0\right\rangle, \ldots, \left|1 \overset{n)}{\ldots} 1\right\rangle\}$, en donde $\left|0 \overset{n)}{\ldots} 0\right\rangle$ es la notación abreviada de $|0\rangle_1 \otimes \overset{n)}{\ldots} \otimes |0\rangle_n$.

Por conveniencia de notación, a veces en lugar de ceros y unos utilizaremos notación numérica $\{|i\rangle\}_{i=0,\ldots,2^n-1}$ para representar los 2^n vectores de la base ortonormal de un sistema de n *qubits*.

En este punto en el que hemos introducido el producto tensorial como herramienta para construir espacios de dimensión mayor, es importante hacer una precisión fundamental para entender por qué funcionan los algoritmos cuánticos. Si bien hay vectores del sistema de n *qubits* que corresponden a productos tensoriales de los vectores individuales de los sistemas de un *qubit*, no todos se pueden descomponer de esa manera. Aquellos vectores que no se pueden construir como productos tensoriales merecen especial atención en computación cuántica, por lo que los identificaremos con un nombre específico.

Definición 2.14. Dado un sistema de n *qubits*, se llama **estado entrelazado** (*entangled state*, en inglés) a un vector que no se puede expresar como producto tensorial de n vectores de cada uno de los subsistemas de un *qubit*.

Dado un sistema de dos *qubits*, los siguientes cuatro vectores, los llamados **estados de Bell**, constituyen un ejemplo muy conocido de estados entrelazados que, además, son una base del espacio:

$$\left\{\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle), \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle), \frac{1}{\sqrt{2}}(|10\rangle + |01\rangle), \frac{1}{\sqrt{2}}(|10\rangle - |01\rangle)\right\} \tag{2.5}$$

Se puede demostrar fácilmente que los anteriores son estados entrelazados: al igualar cualquiera de ellos a una expresión de tipo $(\lambda_0|0\rangle + \lambda_1|1\rangle) \otimes (\mu_0|0\rangle + \mu_1|1\rangle)$, se llega a una contradicción. Es en este tipo de estados entrelazados donde radican las capacidades de la computación cuántica que van más allá de la computación clásica, por lo que su uso es imprescindible en los algoritmos cuánticos sin equivalente clásico.

2.4 Las matrices de Pauli y el grupo de Pauli $\mathcal{P}_n$ para n *qubits*

Introducimos ahora cuatro matrices de gran importancia.

Definición 2.15. Se conocen como **matrices de Pauli** las cuatro matrices complejas 2×2 siguientes[1]:

$$\begin{aligned} I &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \qquad \sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \\ \sigma_y &= \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \qquad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \end{aligned} \tag{2.6}$$

Estas matrices son muy útiles porque tienen varias características importantes, que vamos a ver a continuación.

Proposición 2.5. *Toda matriz compleja* 2×2 *puede expresarse como combinación lineal de las matrices de Pauli.*

Demostración. Sea una matriz compleja arbitraria:

$$M = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \qquad a, b, c, d \in \mathbb{C} \tag{2.7}$$

Igualándola a una combinación lineal de las cuatro matrices y resolviendo el sistema de ecuaciones, se encuentra que la descomposición es la siguiente:

$$M = \frac{a+d}{2} I + \frac{b+c}{2} \sigma_x + \frac{b-c}{2} i\sigma_y + \frac{a-d}{2} \sigma_z \tag{2.8}$$

□

[1] Muchos autores no incluyen la identidad como una auténtica matriz de Pauli, pero resulta conveniente para muchas definiciones y resultados tratarla conjuntamente con las tres matrices popularizadas por Wolfgang Pauli.

Proposición 2.6. *Las matrices de Pauli son inversas de sí mismas, hermíticas y unitarias.*

Demostración. Se demuestra de manera inmediata multiplicando σ_x, σ_y y σ_z consigo mismas y calculando las adjuntas. □

Definición 2.16. En un sistema de un *qubit* fijada una base, se llama **operadores de Pauli** a los endomorfismos cuyas matrices coordenadas son las de Pauli.

Proposición 2.7. *El conjunto de los operadores de Pauli multiplicados por* ± 1 *y* $\pm i$ *tiene estructura de grupo. A este grupo se lo denomina* ***grupo de Pauli*** $\mathcal{P}_1$.

Demostración. El conjunto es cerrado por la operación producto de matrices, cumple la propiedad asociativa, tiene elemento neutro I y cada elemento tiene inverso dentro del conjunto. □

Este grupo de endomorfismos de un sistema de un *qubit* puede extenderse a sistemas de n *qubits* mediante productos tensoriales. Esto da lugar a un concepto de grupo de operadores de Pauli más general, que definimos a continuación y que resultará sumamente útil para el estudio de los endomorfismos unitarios que pueden afectar a los sistemas cuánticos.

Definición 2.17. En un sistema de $n > 1$ *qubits* se llaman **operadores de Pauli sobre** n ***qubits*** a los productos tensoriales de los operadores de Pauli en un *qubit*.

Proposición 2.8. *El conjunto de todos los operadores de Pauli sobre n qubits multiplicados por los factores* ± 1 *y* $\pm i$ *tiene estructura de grupo. A este grupo se lo denomina* ***grupo de Pauli*** $\mathcal{P}_n$.

Demostración. Como en el caso de un *qubit*, el conjunto es cerrado por la operación de grupo, se verifica la propiedad asociativa, hay elemento neutro $I \otimes \cdots \otimes I$ y cada elemento del conjunto tiene inverso. □

Proposición 2.9. *Los operadores del grupo de Pauli* $\mathcal{P}_n$ *solamente admiten dos posibles autovalores:* ± 1.

Demostración. Desarrollando la ecuación característica para las cuatro matrices de Pauli, se obtiene el autovalor 1 para I y el par de autovalores ± 1 para σ_x, σ_y y σ_z. Esto confirma el resultado para el grupo $\mathcal{P}_1$. Para $n > 1$, basta con tener en cuenta la propiedad de que los autovalores de productos tensoriales de aplicaciones lineales son los productos de los autovalores, por lo que solamente podrán valer ± 1 cualquiera que sea n. □

Proposición 2.10. *Dado un operador de Pauli sobre n qubits* $P \in \mathcal{P}_n$, *se cumple o bien* $P^2 = I$ *o bien* $P^2 = -I$.

Demostración. En el caso del grupo $\mathcal{P}_1$, los dieciséis operadores tienen las matrices de Pauli como matrices coordenadas multiplicadas por los cuatro factores posibles ± 1, $\pm i$. Llevando a cabo el producto de matrices, se comprueba que se cumplen las relaciones $\sigma_x^2 = I$, $\sigma_y^2 = I$ y $\sigma_z^2 = I$, por lo que se tiene $(\pm I)^2 = I$, $(\pm\sigma_x)^2 = I$, $(\pm\sigma_y)^2 = I$, $(\pm\sigma_z)^2 = I$ y $(\pm iI)^2 = -I$, $(\pm i\sigma_x)^2 = -I$, $(\pm i\sigma_y)^2 = -I$, $(\pm i\sigma_z)^2 = -I$, con lo que queda demostrado el resultado para $\mathcal{P}_1$.

El resultado para $\mathcal{P}_1$ se extiende por productos tensoriales a todo grupo $\mathcal{P}_n$ con $n > 1$. □

Definición 2.18. Dado un sistema de n *qubits* y un endomorfismo P del grupo $\mathcal{P}_n$ de Pauli, se llama **peso de Pauli** de P al número de componentes del producto tensorial en que se descompone P que son distintas de la identidad.

Así, si por ejemplo, en un sistema de tres *qubits* el operador de Pauli $\sigma_x \otimes I \otimes I$ tiene peso de Pauli 1 mientras que $\sigma_x \otimes \sigma_z \otimes I$ tiene peso 2.

2.5 Una caracterización más potente de los estados de un sistema cuántico: el operador de densidad

Las ideas matemáticas descritas hasta este punto son suficientes para la descripción de los circuitos y algoritmos cuánticos, que veremos más adelante. Pero para poder estudiar los errores a los que se puede ver expuesto un sistema de *qubits*, que abordaremos en la última parte del trabajo, es necesario extender el formalismo mediante el concepto de operadores de densidad, que presentaremos brevemente a continuación.

Hasta ahora hemos asumido que es posible conocer con exactitud los estados que adopta un sistema cuántico; esto es, que tiene sentido experimental considerar que un sistema cuántico queda descrito por un espacio de Hilbert H complejo de dimensión finita d con una base ortonormal $\{|e_i\rangle\}_{i=1,\dots,d}$, de tal modo que el estado del sistema es un vector unitario concreto $\sum_{i=1}^{d} \alpha_i |e_i\rangle$ de este espacio. Pero en la realidad, el experimentador nunca podrá alcanzar ese extremo ideal de certidumbre. A la incertidumbre esencial cuántica implícita en la medición de observables, tenemos que añadir la incertidumbre derivada de dos circunstancias: por un lado, nuestro conocimiento del estado tendrá que describirse como una mezcla estadística o *ensemble* de estados posibles y, por otro, ningún sistema se encuentra totalmente aislado, por lo que el espacio de Hilbert H considerado no es sino una parte de lo que realmente es $H \otimes H_{\text{resto del universo}}$ [21, p. 99] [12].

Para poder estudiar los errores potenciales en computación cuántica necesitamos generalizar el concepto de estados como vectores de un espacio de Hilbert a una idea más amplia de estado que incluya también a estos *ensembles* estadísticos de sistemas. Comenzaremos estableciendo algunas definiciones. Para mayor claridad, utilizaremos ahora preferentemente la designación de "estados puros", que ya habíamos introducido, en lugar de la más simple de "estados".

Definición 2.19. Dado un espacio de estados H, llamamos **estado mixto** a una mezcla estadística de estados puros; es decir, a un conjunto $\{|\psi_i\rangle\}_{i\in I}$ de estados puros (siendo I un conjunto de índices que podemos considerar finito), cada uno de los cuales tiene asociada una probabilidad p_i, cumpliéndose $\sum_{i\in I} p_i = 1$.

Definición 2.20. Dado un espacio de estados puros H, llamamos **espacio de estados mixtos** al conjunto de todos los estados mixtos que se pueden formar con los estados puros, incluidos estos últimos como caso particular.

Para poder describir bajo un modelo uniforme tanto los estados puros como los mixtos, resulta conveniente introducir una formulación de los estados del sistema diferente de la que hemos construido a partir de vectores del espacio de Hilbert. En la nueva formulación, los estados vendrán caracterizados por endomorfismos a los que se denomina "operadores de densidad". Introduciremos primero este lenguaje alternativo para los estados puros ya conocidos y veremos a continuación cómo se extiende a los nuevos estados mixtos. Para ello, necesitaremos algunas definiciones previas.

Definición 2.21. Dado un espacio de estados H, se puede definir un producto externo que hace corresponder a cada par de vectores del espacio $(|\psi\rangle, |\phi\rangle)$ un endomorfismo $O_{\psi,\phi} : H \to H$ definido de manera que: $O_{\psi,\phi}|x\rangle = \langle\phi|x\rangle|\psi\rangle$. En la notación *bra-ket* de Dirac, a este endomorfismo se lo representa como $|\psi\rangle\langle\phi|$.

Definición 2.22. Dado un vector $|\psi\rangle$ de un espacio de estados H, llamamos **operador proyector sobre** $|\psi\rangle$ al endomorfismo en H que resulta de aplicar el producto externo de la definición anterior (2.21) sobre el mismo vector $|\psi\rangle$; en notación *bra-ket*: $|\psi\rangle\langle\psi|$.

El nombre de "proyector" se justifica por el hecho de que cuando el vector $|\psi\rangle$ es unitario se cumple la relación $(|\psi\rangle\langle\psi|)^2 = |\psi\rangle\langle\psi|$, característica de las proyecciones.

Otra noción importante es la de "traza" de un endomorfismo:

Definición 2.23. Dado un endomorfismo P sobre un espacio de estados H de dimensión finita d, con base $\{|i\rangle\}_{i=0,\ldots,d-1}$, se llama **traza** de P, $Tr(P)$, al número real $\sum_{i=0}^{d-1} \langle i|P|i\rangle$.

Esta definición de la traza equivale a la definición basada en la representación matricial como "suma de los elementos de la diagonal de la matriz". Además, se puede demostrar fácilmente que no depende de la elección de base $|i\rangle$ (basta con desarrollar en dos bases ortonormales diferentes y tener en cuenta que la transformación entre las dos bases es una aplicación unitaria).

Por otra parte, hemos visto que el valor esperado de un operador hermítico P en un espacio de Hilbert H cuando el sistema se encuentra en el estado $|\psi\rangle$ viene dado por $\langle\psi|P|\psi\rangle$. Esta expresión puede reformularse a partir del operador proyector, de la manera siguiente:

Proposición 2.11. *Dado un espacio de estados H en el que se define un endomorfismo hermítico P, para todo $|\psi\rangle \in H$ se cumple: $\langle\psi|P|\psi\rangle = Tr(P|\psi\rangle\langle\psi|) = Tr(|\psi\rangle\langle\psi|P)$.*

Demostración. Si d es la dimensión de H, dada una base $\{|i\rangle\}_{i=0,\dots,d-1}$ observemos que la suma de los proyectores de la base es igual al endomorfismo unidad; esto es, $I = \sum_{i=0}^{d-1} |i\rangle\langle i|$. Introduciendo la identidad bajo esta forma en medio de la expresión $\langle\psi|P|\psi\rangle$ y teniendo en cuenta que, por (2.2), $\langle\psi|P|\psi\rangle$ es un número real y que P es un endomorfismo hermítico ($P = P^\dagger$), podemos desarrollar la expresión de la siguiente manera:

$$\begin{aligned} \langle\psi|P|\psi\rangle = \langle\psi| \left(\sum_{i=0}^{d-1} |i\rangle\langle i| \right) P|\psi\rangle = \sum_{i=0}^{d-1} \langle\psi|i\rangle\langle i|P|\psi\rangle = \\ = \sum_{i=0}^{d-1} \overline{\langle i|\psi\rangle}\ \overline{\langle\psi|P^\dagger|i\rangle} = \sum_{i=0}^{d-1} \overline{\langle i|\psi\rangle\langle\psi|P^\dagger|i\rangle} = \sum_{i=0}^{d-1} \langle i|\psi\rangle\langle\psi|P^\dagger|i\rangle = \\ = \sum_{i=0}^{d-1} \langle i|\psi\rangle\langle\psi|P|i\rangle = Tr(|\psi\rangle\langle\psi|P) \end{aligned} \tag{2.9}$$

Análogamente, reemplazando $\langle\psi|P|\psi\rangle$ por $\langle\psi|P(\sum_{i=0}^{d} |i\rangle\langle i|)|\psi\rangle$ se llega a la otra igualdad.

□

De esta manera, hemos introducido una formulación diferente de los estados de un sistema físico y de los valores esperados de los observables del sistema. Pero ¿qué se gana con este enfoque alternativo? Si ya habíamos caracterizado los estados de un sistema cuántico como vectores *ket* (ignorando el factor de fase físicamente irrelevante), ¿qué aporta tratarlos como proyectores $|\psi\rangle\langle\psi|$? Y si ya sabíamos cómo calcular valores esperados de un endomorfismo hermítico P que representa una magnitud física observable mediante el cálculo de $\langle\psi|P|\psi\rangle$, ¿qué nos aporta calcularlo ahora como $Tr(|\psi\rangle\langle\psi|P)$? La respuesta, como ya hemos sugerido, radica en los estados mixtos. Veremos a continuación cómo esta nueva formulación se extiende de manera natural a los estados mixtos, permitiéndonos además diferenciar estados mixtos de puros.

Para ver cómo extender el concepto de estado puro a un estado mixto, supongamos que disponemos de un sistema físico cuyo estado exacto desconocemos, pero que podemos modelar como una mezcla estadística en la cual habría n diferentes estados posibles $\{|\psi_i\rangle\}_{i=1,\dots,n}$, cada uno de los cuales tiene una probabilidad p_i de ser el estado en el que se encuentra realmente el sistema. Entonces el valor esperado de un observable P vendrá dado por $\sum_{i=1}^{n} p_i\langle\psi_i|P|\psi_i\rangle$ o, en la nueva formulación, $\sum_{i=1}^{n} p_i Tr(|\psi_i\rangle\langle\psi_i|P)$. Utilizando la notación $\langle P\rangle$ para el valor esperado y teniendo en cuenta que la traza es una aplicación lineal, podemos desarrollar esta última

expresión:

$$\begin{aligned}\langle P\rangle = \sum_{i=1}^{n} p_i Tr(|\psi_i\rangle\langle\psi_i|P) = Tr\left(\sum_{i=1}^{n} p_i|\psi_i\rangle\langle\psi_i|P\right) = \\ = Tr\left(\left(\sum_{i=1}^{n} p_i|\psi_i\rangle\langle\psi_i|\right)P\right)\end{aligned} \tag{2.10}$$

En esta última expresión vemos que en un estado mixto el endomorfismo hermítico $\sum_{i=1}^{n} p_i|\psi_i\rangle\langle\psi_i|$ desempeña exactamente el mismo papel que el operador proyector para los estados puros. Dada su utilidad, podemos entonces dar un nombre a esta aplicación.

Definición 2.24. Dado un conjunto de estados mixtos, se llama **operador de densidad**, o simplemente **matriz de densidad**, de un estado del sistema al operador proyector $|\psi\rangle\langle\psi|$ cuando se trata de un estado puro $|\psi\rangle$ y a la combinación lineal de operadores proyectores $\sum_{i=1}^{n} p_i|\psi_i\rangle\langle\psi_i|$ cuando se trata de una mezcla estadística de n estados puros $\{|\psi_i\rangle\}_{i=0,\dots,n}$, cada uno de ellos con probabilidad respectiva p_i. Al operador de densidad se lo designa habitualmente mediante la letra ρ.

Armados con esta definición, podemos entonces describir sistemas mixtos y sistemas puros bajo un mismo paraguas formal. Veamos en primer lugar una propiedad importante del operador de densidad.

Proposición 2.12. *Sea ρ un operador de densidad. Se cumple: $Tr(\rho) = 1$.*

Demostración. Supongamos primero que ρ describe un estado puro. Entonces existirá un vector de estado $|\psi\rangle$ tal que se cumple $\rho = |\psi\rangle\langle\psi|$. Si el espacio de Hilbert tiene dimensión d y tiene una base ortonormal $\{|i\rangle\}_{i=0,\dots,d-1}$, se tiene que $|\psi\rangle = \sum_{i=0}^{d-1} a_i|i\rangle$, cumpliéndose la condición de normalización $\sum_{i=0}^{d-1}|a_i|^2 = 1$ y entonces:

$$\begin{aligned}Tr(\rho) = \sum_{i=0}^{d-1}\langle i|\psi\rangle\langle\psi|i\rangle = \sum_{i=0}^{d-1}\langle i|\left(\sum_{j=0}^{d-1} a_j|j\rangle\right)\left(\sum_{j=0}^{d-1}\overline{a_j}\langle j|\right)|i\rangle = \\ \sum_{i=0}^{d-1}\left(\sum_{j=0}^{d-1} a_j\langle i|j\rangle\right)\left(\sum_{j=0}^{d-1}\overline{a_j}\langle j|i\rangle\right) = \sum_{i=0}^{d-1} a_i\overline{a_i} = \sum_{i=0}^{d-1}|a_i|^2 = 1.\end{aligned} \tag{2.11}$$

Si, en cambio, ρ describe un estado mixto, entonces existirán $k \in \mathbb{N}$ estados puros $\{\rho_i\}_{i=0,\dots,k}$ de modo que $\rho = \sum_{i=0}^{k} p_i\rho_i$ siendo cada ρ_i un estado puro (luego, por el resultado anterior, $Tr(\rho_i) = 1$) y cada p_i una probabilidad ($p_i \in [0,1]$), de modo que $\sum_{i=0}^{k} p_i = 1$. Entonces se tiene:

$$Tr(\rho) = Tr(\sum_{i=0}^{k} p_i \rho_i) = \sum_{i=0}^{k} p_i Tr(\rho_i) = \sum_{i=0}^{k} p_i = 1 \tag{2.12}$$

□

La propiedad anterior es común a estados puros y mixtos, pero podemos obtener un criterio característico de cada tipo de estado si investigamos las propiedades de ρ^2.

Lema 2.13. *Sea ρ el operador densidad correspondiente a un estado puro. Entonces se cumple: $Tr(\rho^2) = 1$.*

Demostración. Al ser un estado puro, ρ cumple la relación de proyección o idempotencia $\rho^2 = \rho$. Y por la proposición (2.12):

$$Tr(\rho^2) = Tr(\rho) = 1 \tag{2.13}$$

□

Pero si el estado es mixto no puro, entonces $\rho^2 \neq \rho$ y se puede demostrar que $Tr(\rho^2) < 1$. Enunciaremos este resultado como una proposición que nos da una caracterización de la naturaleza, pura o mixta, de un estado:

Proposición 2.14. *Sea ρ el operador densidad correspondiente a un estado mixto, que consideramos que incluye como caso particular la posibilidad de que sea puro. Entonces se cumple: $Tr(\rho^2) \leqslant 1$. Y, además, $Tr(\rho^2) = 1$ si y solo si el estado es puro.*

Demostración. Si ρ es el operador de densidad de un estado puro, sabemos por el lema (2.13) que se cumple $Tr(\rho^2) = 1$. Si ρ es el operador de densidad de un estado mixto, entonces habrá un conjunto de $h \in \mathbb{N}$ estados puros con operadores de densidad $\{\rho_i\}_{i=1,\dots,h}$ de modo que se cumple $\rho = \sum_{i=1}^{k} p_i \rho_i$, en donde $\{p_i\}_{i=1,\dots,h}$ son h probabilidades (es decir, valores reales entre 0 y 1) que cumplen $\sum_{i=1}^{h} p_i = 1$. Cada uno de estos estados puros ρ_i equivaldrá a $|\psi_i\rangle\langle\psi_i|$, donde $|\psi_i\rangle$ es un vector del espacio de Hilbert H.

Si el espacio de Hilbert H tiene dimensión d y una base ortonormal $\{|0\rangle, \dots, |d-1\rangle\}$, cada vector $|\psi_i\rangle$ puede expresarse como: $|\psi_i\rangle = \sum_{j=0}^{d-1} \alpha_{ij}|j\rangle$, en donde las coordenadas $\{\alpha_{ij}\}$ cumplen $\sum_{j=0}^{d-1} |\alpha_{ij}|^2 = 1$. Luego el operador de densidad se puede expresar como:

$$\rho = \sum_{i=1}^{h} p_i |\psi_i\rangle\langle\psi_i| = \sum_{i=1}^{h} p_i \left(\sum_{j=0}^{d-1} \alpha_{ij}|j\rangle\right)\left(\sum_{k=0}^{d-1} \overline{\alpha_{ik}}\langle k|\right) = \\ = \sum_{i=1}^{h}\sum_{j=0}^{d-1}\sum_{k=0}^{d-1} p_i \alpha_{ij}\overline{\alpha_{ik}}|j\rangle\langle k| \tag{2.14}$$

Utilizando esta última expresión para calcular ρ^2 (teniendo en cuenta que $|j\rangle\langle j|j\rangle\langle j|$ $= |j\rangle\langle j|$ y que $|j\rangle\langle j|k\rangle\langle k| = 0$ para $j \neq k$), se tiene:

$$\rho^2 = \sum_{i=1}^{h}\sum_{j=0}^{d-1}\sum_{k=0}^{d-1}\sum_{l=1}^{h}\sum_{m=0}^{d-1}\sum_{n=0}^{d-1} p_i p_l \alpha_{ij}\overline{\alpha_{ik}}\alpha_{lm}\overline{\alpha_{ln}}|j\rangle\langle k|m\rangle\langle n| = \\ = \sum_{i=1}^{h}\sum_{j=0}^{d-1}\sum_{k=0}^{d-1}\sum_{l=1}^{h}\sum_{n=0}^{d-1} p_i p_l \alpha_{ij}\overline{\alpha_{ik}}\alpha_{lk}\overline{\alpha_{ln}}|j\rangle\langle n| \tag{2.15}$$

Y tomando la traza y teniendo en cuenta la linealidad y que $Tr(|j\rangle\langle j|) = 1$ y $Tr(|j\rangle\langle n|) = 0$ para $j \neq n$:

$$Tr(\rho^2) = \sum_{i=1}^{h}\sum_{j=0}^{d-1}\sum_{k=0}^{d-1}\sum_{l=1}^{h} p_i p_l \alpha_{ij}\overline{\alpha_{ik}}\alpha_{lk}\overline{\alpha_{lj}} = \\ = \sum_{i=1}^{h}\sum_{l=1}^{h} p_i p_l \left(\sum_{j=0}^{d-1}\overline{\alpha_{lj}}\alpha_{ij}\right)\left(\sum_{k=0}^{d-1}\overline{\alpha_{ik}}\alpha_{lk}\right) = \\ = \sum_{i=1}^{h}\sum_{l=1}^{h} p_i p_l \langle\psi_l|\psi_i\rangle\langle\psi_i|\psi_l\rangle = \sum_{i=1}^{h}\sum_{l=1}^{h} p_i p_l \overline{\langle\psi_i|\psi_l\rangle}\langle\psi_i|\psi_l\rangle = \\ = \sum_{i=1}^{h}\sum_{l=1}^{h} p_i p_l |\langle\psi_i|\psi_l\rangle|^2 \tag{2.16}$$

Si se trata de un estado puro, entonces $h = 1$ y recuperamos el resultado del lema anterior, ahora como caso particular. Si, por el contrario, se trata de un estado mixto, entonces hay varios sumandos, $h > 1$. En tal caso, podemos aplicar a (2.16) la desigualdad de Cauchy-Schwarz [22, p. 183], $|\langle\phi|\psi\rangle| \leqslant \sqrt{|\langle\phi|\phi\rangle|}\sqrt{|\langle\psi|\psi\rangle|}$ y tener en cuenta que los estados puros $|\psi_i\rangle$ son vectores unitarios:

$$Tr(\rho^2) \leqslant \sum_{i=1}^{h}\sum_{l=1}^{h} p_i p_l |\langle\psi_i|\psi_i\rangle||\langle\psi_l|\psi_l\rangle| = \\ = \sum_{i=1}^{h}\sum_{l=1}^{h} p_i p_l = \left(\sum_{i=1}^{h} p_i\right)\left(\sum_{l=1}^{h} p_l\right) = 1 \tag{2.17}$$

La desigualdad anterior (2.17) es estricta para los estados mixtos no puros, ya que en tal caso los vectores $|\psi_i\rangle$ y $|\psi_l\rangle$ con índices distintos corresponden a estados puros diferentes que no pueden ser uno múltiplo del otro, por lo que tendremos $Tr(\rho^2) < 1$.

□

El criterio que acabamos de obtener nos permite definir una medida de cuán puro es un estado.

Definición 2.25. Dado un estado descrito por un operador de densidad ρ, se llama **pureza** del estado a la función real $\gamma(\rho) := Tr(\rho^2)$.

Se puede demostrar que la pureza está entre los valores $1/d$ (dimensión del espacio) y 1.

Un concepto similar es el de la entropía, que en la teoría de la información cuántica puede definirse a partir de la pureza:

Definición 2.26. Dado un estado de un sistema cuántico descrito por un operador de densidad ρ, se llama **entropía lineal** del estado al número real $1 - \gamma(\rho)$.

Los valores posibles de la entropía, tal como la hemos definido, están en el intervalo $[0, \frac{d-1}{d}]$. A veces se incluye en la definición de entropía un factor de normalización $\frac{d-1}{d}$ para que el intervalo de valores posibles sea $[0, 1]$.

Habíamos visto que un estado puro evoluciona en un intervalo de tiempo por la aplicación de un endomorfismo unitario U sobre el vector de estado. Es decir, si el sistema cuántico se encuentra inicialmente en el estado representado por un vector $|\psi\rangle$, entonces en un instante de tiempo posterior su estado pasará a ser $U|\psi\rangle$. A partir de la definición de operador de densidad para los estados puros, tenemos que el estado evoluciona según la siguiente ley:

$$\rho = |\psi\rangle\langle\psi| \mapsto U|\psi\rangle\langle\psi|U^\dagger = U\rho U^\dagger \tag{2.18}$$

La expresión (2.18) resulta ser también válida para los estados mixtos. Este resultado fundamental es el contenido de la siguiente proposición:

Proposición 2.15. *Sea un sistema de estados mixtos en el que está definida una evolución temporal con dos instantes de tiempo, un estado inicial definido por un operador de densidad ρ_0 y un estado final definido por otro operador de densidad ρ_1. Si cada estado puro inicial descrito por un vector $|\psi_0\rangle$ evoluciona, mediante la transformación debida a un endomorfismo unitario U, hasta un estado puro final $|\psi_1\rangle = U|\psi_0\rangle$, entonces para todo estado mixto (incluidos los puros como caso particular) se cumplirá: $\rho_1 = U\rho_0 U^\dagger$.*

Demostración. Hemos demostrado ya la relación para el caso particular de los es-

tados puros. En el caso de un estado mixto, basta con considerarlo formado por un *ensemble* de estados puros representados en el estado inicial por un conjunto de vectores $\{|\psi_{0,i}\rangle\}_{i=1,\ldots,n}$, cada uno con probabilidad p_i. Los estados finales correspondientes serán $\{|\psi_{1,i}\rangle\}_{i=1,\ldots,n} = \{U|\psi_{0,i}\rangle\}_{i=1,\ldots,n}$. A partir de la expresión para el operador de densidad de un estado mixto, tendremos para el estado inicial:

$$\rho_0 = \sum_{i=1}^{n} p_i |\psi_{0,i}\rangle\langle\psi_{0,i}| \tag{2.19}$$

Y para el estado final:

$$\begin{aligned} \rho_1 = \sum_{i=1}^{n} p_i |\psi_{1,i}\rangle\langle\psi_{1,i}| = \sum_{i=1}^{n} p_i U|\psi_{0,i}\rangle\langle\psi_{0,i}|U^\dagger = \\ U\left(\sum_{i=1}^{n} p_i |\psi_{0,i}\rangle\langle\psi_{0,i}|\right)U^\dagger = U\rho_0 U^\dagger \end{aligned} \tag{2.20}$$

□

En el libro de Michael A. Nielsen y Isaac L. Chuang [21], una de las obras de referencia más influyentes en el campo de la computación cuántica, a la aplicación ϵ que lleva ρ a $\epsilon(\rho) = U\rho U^\dagger$ se le da el nombre "operación cuántica", que se aplica también en general a otras formas de evolución de un sistema cuántico como la medición. En el primer caso, transformaciones unitarias, se habla de "operaciones cuánticas que conservan la traza" (*trace-preserving quantum operations*) y en el segundo, mediciones, "operaciones cuánticas que no conservan la traza" (*non-trace-preserving quantum operations*). Esta idea de operación cuántica es debida a Karl Kraus [16] y permite agrupar como dos casos particulares de un concepto general las operaciones de medición y de transformación unitaria. Aunque no utilizaremos esta terminología en el contexto del presente trabajo, conviene conocerla al ser muy habitual en la literatura sobre códigos correctores.

Un último resultado interesante para los estados mixtos, al que ya habíamos aludido sin entrar en detalle, es el hecho de que este tipo de estados más generales no solamente aparecen como consecuencia de un conocimiento imperfecto del estado (puro) de un sistema, sino que surgen también de manera natural como la manifestación en un subsistema de un estado puro perfectamente conocido en un sistema total mayor. ¿En qué caso puede ocurrir esto? Habíamos visto que dados dos sistemas cuánticos, podemos formar un sistema compuesto mediante producto tensorial de los espacios. Además, hay estados del sistema compuesto que se forman como producto tensorial de estados concretos de los sistemas componentes; en tal caso, esos estados que por producto tensorial forman el estado del sistema mayor serán los estados de los subsistemas. Pero habíamos visto que hay también un tipo de estados, los estados entrelazados, que no admiten esa descomposición. ¿Será posible

de alguna manera expresar los estados de los subsistemas cuando el estado total es entrelazado? Los estados mixtos entran aquí en escena como respuesta a esta pregunta. Para ver cómo se manifiesta esto, es necesario definir el concepto del estado de un subsistema, algo que podemos hacer de manera elegante con el formalismo de operadores de densidad.

Sea un sistema cuántico que consta de $n \in \mathbb{N}$ *qubits*. Supongamos que deseamos considerar por separado dos grupos de *qubits*. Por un lado, $m \in \mathbb{N}, m < n$ *qubits* forman un subsistema que llamaremos **sistema principal** y los $n-m$ *qubits* restantes forman lo que llamaremos **sistema del entorno**. Básicamente, estamos dividiendo un sistema grande en una parte que tiene interés especial y el resto del universo con el que potencialmente interactúa. En la situación descrita, el espacio de estados del sistema total será un espacio de Hilbert H de dimensión 2^n, mientras que el espacio de Hilbert H_0 del subsistema principal tendrá dimensión 2^m y el espacio de Hilbert H_{ent} del subsistema del entorno tendrá dimensión 2^{n-m}. Tal como hemos visto anteriormente, el espacio total será igual al producto tensorial de los espacios para los subsistemas $H = H_0 \otimes H_{ent}$.

Para poder definir el estado de un subsistema principal en este tipo de planteamiento habitual en el análisis de errores, es necesario introducir primero una nueva operación matemática:

Definición 2.27. Sean dos sistemas cuánticos H_A, de n_A *qubits*, y H_B, de n_B *qubits*. Sea H_{AB} el sistema de $n_A + n_B$ *qubits* producto tensorial de los sistemas H_A y H_B. Se llama **traza parcial** del sistema H_A sobre el sistema H_B a una aplicación Tr_B definida en H_{AB} que cumple las dos propiedades siguientes:

1. Tr_B es una aplicación lineal.

2. Dados dos estados puros $|a_1\rangle$ y $|a_2\rangle$ de H_A y dos estados puros $|b_1\rangle$ y $|b_2\rangle$ de H_B, se cumple $Tr_B(|a_1\rangle\langle a_2| \otimes |b_1\rangle\langle b_2|) = |a_1\rangle\langle a_2|\langle b_2|b_1\rangle$

$\langle b_2|b_1\rangle$ es igual a $Tr(|b_1\rangle\langle b_2|$, donde Tr es la operación de traza convencional definida en H_B, lo que justifica su nombre. Por linealidad, la definición para productos tensoriales de estados puros determina el valor de la traza parcial para los estados entrelazados y mixtos del sistema total H_{AB}. Esta definición de traza parcial sirve para dar una definición de operador de densidad de un subsistema.

Definición 2.28. Sean dos sistemas cuánticos H_A, de n_A *qubits*, y H_B, de n_B *qubits*. Sea H_{AB} el sistema de $n_A + n_B$ *qubits* producto tensorial de los sistemas H_A y H_B. Si el estado del sistema H_{AB} viene descrito por un operador de densidad ρ^{AB}, se llama **operador de densidad reducido** del sistema H_A en el sistema total H_{AB} al endomorfismo definido en H_{AB} por la expresión $\rho^A := Tr_B(\rho^{AB})$.

Con esto concluimos esta introducción al formalismo matemático de los sistemas cuánticos. Hemos definido así tanto la manera en que se componen sistemas de *qubits* para formar sistemas más grandes como también la manera en que se puede restringir la descripción a un subsistema partiendo del sistema mayor. Como último

resultado, estamos ya en disposición de poder comprobar que un estado entrelazado de un sistema se manifiesta como estado mixto en un subsistema. No haremos una demostración general, sino que nos conformaremos con captar la idea analizando un caso particular muy sencillo, el del estado de Bell $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ que habíamos visto como ejemplo de estado entrelazado en (2.5).

Si H es el espacio de Hilbert de un sistema de un *qubit* y $H^{\otimes 2}$ es el del sistema de dos *qubits* obtenido como producto tensorial de H por sí mismo, supongamos $H^{\otimes 2}$ se encuentra en el estado puro entrelazado $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$. Entonces el estado expresado como operador de densidad será el siguiente:

$$\rho^{AB} = \frac{1}{2}(|00\rangle\langle 00| + |00\rangle\langle 11| + |11\rangle\langle 00| + |11\rangle\langle 11|)$$

Para tratar el sistema del primer *qubit* dejando de lado el estado del segundo, será necesario recurrir a la definición (2.28) para el operador de densidad de un subsistema, con lo que tendremos un operador de densidad reducido al primer *qubit* ρ^A [21, p. 106] (utilizando las letras A y B de la definición general, que identifican en este caso al primer y segundo *qubit*, respectivamente):

$$\rho^A = Tr_B(\rho^{AB}) = Tr_B(\frac{1}{2}(|00\rangle\langle 00| + |00\rangle\langle 11| + |11\rangle\langle 00| + |11\rangle\langle 11|)) =$$
$$= \frac{1}{2}(Tr_B(|00\rangle\langle 00|) + Tr_B(|00\rangle\langle 11|) + Tr_B(|11\rangle\langle 00|) + Tr_B(|11\rangle\langle 11|))) = \frac{1}{2}I$$

Y este estado es un estado mixto ya que la traza de su cuadrado es $Tr((\rho^A)^2) = 1/2$. Se confirma así la idea antes adelantada de que los estados mixtos describen también el efecto sobre un subsistema de los estados entrelazados del sistema total.

3 Las puertas lógicas cuánticas

Una vez definido el marco formal de los *qubits*, ¿cómo los utilizaremos para llevar a cabo operaciones lógicas? De la misma manera que en la computación clásica se somete a los bits a transformaciones haciéndolos pasar por puertas lógicas, para poder desarrollar una teoría de la computación cuántica necesitaremos aplicar transformaciones análogas a los *qubits*. Hemos visto que las leyes de la naturaleza permiten que un sistema de *qubits* se transforme mediante operaciones de medición y mediante endomorfismos unitarios. Son precisamente estos últimos las transformaciones que permiten hacer operaciones lógicas con *qubits*. Para ver por qué, pensemos en el caso de la puerta lógica clásica NOT.

La puerta lógica NOT es la más sencilla, no trivial, de las puertas lógicas clásicas. Dado un bit b; al aplicarle una puerta NOT, b cambia de valor pasando de 0 a 1

y de 1 a 0. Para poder realizar este tipo de operación tan sencilla sobre un *qubit*, necesitaremos una operación que transforme el *qubit* $|0\rangle$ en $|1\rangle$ y el *qubit* $|1\rangle$ en $|0\rangle$. Es evidente que este tipo de transformación no puede ser una operación de medición, ya que en ese caso la medición de un estado de la base, $|0\rangle$ o $|1\rangle$, mantendría el estado tal cual. Nos queda entonces el otro tipo de transformación admisible en un sistema cuántico: los endomorfismos unitarios.

Encontrar un endomorfismo unitario que cumpla la condición definitoria de una puerta NOT es muy sencillo. En el Apéndice 1 se demuestra la forma general de una matriz unitaria 2×2 en función de cuatro parámetros reales, que es la siguiente:

$$\begin{gathered} U = \begin{pmatrix} re^{i(\pi+\theta_{10}+\theta_{01}-\theta_{11})} & \sqrt{1-r^2}e^{i\theta_{01}} \\ \sqrt{1-r^2}e^{i\theta_{10}} & re^{i\theta_{11}} \end{pmatrix} \\ r \in [0,1]; \theta_{10}, \theta_{01}, \theta_{11} \in [-\pi, \pi] \end{gathered} \tag{3.1}$$

Escribiendo matricialmente las igualdades $U|0\rangle = |1\rangle$ y $U|1\rangle = |0\rangle$, que debe cumplir el endomorfismo unitario buscado, tenemos:

$$\begin{aligned} \begin{pmatrix} re^{i(\pi+\theta_{10}+\theta_{01}-\theta_{11})} & \sqrt{1-r^2}e^{i\theta_{01}} \\ \sqrt{1-r^2}e^{i\theta_{10}} & re^{i\theta_{11}} \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ \begin{pmatrix} re^{i(\pi+\theta_{10}+\theta_{01}-\theta_{11})} & \sqrt{1-r^2}e^{i\theta_{01}} \\ \sqrt{1-r^2}e^{i\theta_{10}} & re^{i\theta_{11}} \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \end{aligned} \tag{3.2}$$

Las ecuaciones matriciales (3.2) equivalen a cuatro ecuaciones con cuatro incógnitas reales:

$$\begin{aligned} re^{i(\pi+\theta_{10}+\theta_{01}-\theta_{11})} &= 0 \\ \sqrt{1-r^2}e^{i\theta_{10}} &= 1 \\ \sqrt{1-r^2}e^{i\theta_{01}} &= 1 \\ re^{i\theta_{11}} &= 0 \end{aligned} \tag{3.3}$$

Para que se cumplan las ecuaciones primera y cuarta ha de ser $r = 0$. Entonces, la segunda y la tercera ecuación toman la forma siguiente:

$$\begin{aligned} e^{i\theta_{10}} &= 1 \\ e^{i\theta_{01}} &= 1 \end{aligned} \tag{3.4}$$

Por tanto, ha de ser $\theta_{10} = 0$ y $\theta_{01} = 0$. Llevando estos valores a la expresión general de la matriz unitaria 2×2, nos queda:

$$U = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \tag{3.5}$$

La matriz que hemos encontrado es una vieja conocida. Se trata de una de las matrices de Pauli introducidas en la sección anterior.

Así pues, hemos llegado a un resultado importante: la matriz de Pauli σ_x es la matriz coordenada del endomorfismo unitario que representa en un sistema de un *qubit* un papel análogo al de la puerta NOT en un sistema clásico de un bit. Podemos formalizar esta idea con una definición:

Definición 3.1. Dado un sistema de 1 *qubit* H, se llama **puerta lógica cuántica NOT** al endomorfismo unitario cuya matriz coordenada en la base $\{|0\rangle, |1\rangle\}$ es la matriz de Pauli σ_x:

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \tag{3.6}$$

Basándonos en este primer caso, aventuraremos que, en efecto, es posible construir toda una teoría de la computación basada en endomorfismos unitarios. Podemos dar a esta equivalencia postulada entre puertas lógicas y endomorfismos unitarios rango de definición:

Definición 3.2. Dado un sistema de n *qubits* H, se llama **puerta lógica cuántica** a cualquier endomorfismo unitario en H.

Dado que la aplicación reiterada de varias transformaciones unitarias es también una transformación unitaria, la definición anterior implica que la aplicación de sucesivas puertas lógicas es simplemente un producto de transformaciones unitarias y, por tanto, también será una puerta lógica cuántica. Cuando la puerta lógica se puede descomponer en puertas lógicas más simples, hablaremos de "circuitos lógicos cuánticos". Pero independientemente de la idea que se pretenda enfatizar, debe tenerse presente que las tres expresiones "puerta lógica cuántica de n *qubits*", "circuito lógico cuántico de n *qubits*" y "endomorfismo unitario sobre un sistema de n *qubits*" son esencialmente sinónimas de acuerdo con nuestra definición.

Para representar gráficamente los circuitos lógicos utilizaremos a veces diagramas en los que se muestran en el lado izquierdo el estado inicial o "de entrada" (en inglés, *input state*) de los *qubits* que sufren la transformación unitaria y a la derecha el estado final o "de salida" (*output state*) en el que acaban los *qubits* transformados. Las puertas lógicas se representan mediante cajas entre los estados de entrada y de salida. En el caso que hemos introducido de la puerta lógica NOT, representaremos a este endomorfismo unitario con la letra X (podríamos usar σ_x, pero es más habitual escribir simplemente X en el lenguaje de circuitos) y tendremos el diagrama siguiente:

$$\alpha|0\rangle + \beta|1\rangle \text{ ——}\boxed{X}\text{—— } \beta|0\rangle + \alpha|1\rangle$$

En el caso de un único *qubit*, aparece ya una diferencia fundamental con el caso clásico. Mientras que un bit clásico solamente puede sufrir dos transformaciones, que son la identidad y la inversión de valor de la puerta NOT, un sistema de un *qubit* puede sufrir las transformaciones análogas, que serían la identidad y la puerta X, pero hay también toda una familia de endomorfismos unitarios diferentes de estos dos. Jugando con los cuatro parámetros reales de la forma general (3.1) podemos obtener otras posibles puertas lógicas, entre las cuales estarán las otras dos matrices de Pauli, σ_y y σ_z:

$$\sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \qquad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \tag{3.7}$$

Las tres matrices de Pauli pueden interpretarse como rotaciones de ángulo π en torno a tres ejes diferentes en una construcción geométrica llamada "esfera de Bloch", en la que se toman los estados base $|0\rangle$ y $|1\rangle$ como puntos antipodales de una esfera en $\mathbb{R}^3$. ¿Pero tendrán utilidad estos endomorfismos como puertas lógicas? La investigación en computación cuántica que veremos más adelante revela que sí. Y, de hecho, las capacidades sorprendentes de la computación cuántica dependen precisamente de este tipo de transformaciones sin análogo clásico. Adoptando la notación con letra mayúscula habitual en la representación de circuitos cuánticos, conocemos ya entonces tres puertas lógicas X, Y y Z para el sistema de un *qubit*. En realidad, se puede prescindir por lo general de la puerta Y ya que se cumple la relación $Y = iXZ$. Dado que hemos visto que los factores complejos que multiplican a los *qubits* no tienen significado físico, podemos considerar la puerta lógica Y como equivalente a la aplicación consecutiva de Z y X. Las puertas lógicas independientes y no triviales que tenemos son entonces X y Z. Aquí aparece la primera diferencia fundamental entre el sistema clásico de un bit y el cuántico de un *qubit*. Mientras que en el sistema clásico la puerta NOT es la única puerta lógica posible no trivial, en el sistema de un *qubit* tenemos más puertas lógicas posibles, como Z, que no tienen análogo clasico. Veamos el diagrama de una puerta Z solitaria en un sistema de un *qubit*:

$$\alpha|0\rangle + \beta|1\rangle \text{ ——}\boxed{Z}\text{—— } \alpha|0\rangle - \beta|1\rangle$$

Y no se acaban aquí las puertas lógicas interesantes en el sistema de un *qubit*. Otra puerta lógica muy útil es la siguiente, que simplemente introduce un cambio de fase de ángulo θ en la segunda componente:

$$R_\theta = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{pmatrix} \tag{3.8}$$

Esta puerta lógica cuántica no tendría sentido en el mundo clásico, ya que el *qubit* observable $|0\rangle$ se mantiene igual y el *qubit* observable $|1\rangle$ se convierte en $e^{i\theta}|1\rangle$, que decae a $|1\rangle$ con probabilidad 1 cuando es observado, por lo que clásicamente esta puerta lógica equivaldría a la identidad. En computación cuántica, sin embargo, estos cambios de fase pueden aprovecharse de maneras sorprendentes para la paralelización de ciertos cálculos. Dos versiones importantes de las puerta R_θ son las que tienen $\theta = \pi/2$ y $\theta = \pi/4$, a las que a veces se dan los nombres S y T, respectivamente [21, p. 177].

Hay otra puerta lógica cuántica, la puerta de Hadamard, para el sistema de un único *qubit* que aparece reiteradamente en todos los algoritmos cuánticos:

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \tag{3.9}$$

Esta puerta cuántica tiene la peculiaridad de que convierte a los *qubits* observables $|0\rangle$ y $|1\rangle$ en superposiciones que contienen partes iguales de ambos. Por lo tanto, una vez que hemos observado un *qubit* y sabemos que su valor es $|0\rangle$ o $|1\rangle$, basta con aplicarle una puerta de Hadamard para que tengamos una superposición de los dos estados, ambos con probabilidad $\frac{1}{2}$ de ser observados. La relevancia de la puerta de Hadamard es inmensa. Hasta ahora, hemos asumido que sería tecnológicamente factible preparar *qubits* en los estados observables $|0\rangle$ y $|1\rangle$, pero la preparación de estados entrelazados parece en principio problemática. ¿Cómo asegurar que un sistema de un *qubit* se encuentra en un estado entrelazado si no es posible medirlo? La puerta de Hadamard proporciona la solución a este reto tecnológico. Si se consigue construir un dispositivo capaz de preparar un *qubit* en un estado inicial $|0\rangle$ y capaz también de aplicar a este estado inicial una puerta de Hadamard, entonces tendremos resuelto el problema tecnológico de preparar estados con acoplamiento máximo. En la representación esquemática de circuitos cuánticos, la puerta de Hadamard se representa con una letra H mayúscula:

$$|0\rangle \text{ ——} \boxed{H} \text{——} \ \tfrac{1}{\sqrt{2}}|0\rangle + \tfrac{1}{\sqrt{2}}|1\rangle$$

$$|1\rangle \text{ ——} \boxed{H} \text{——} \ \tfrac{1}{\sqrt{2}}|0\rangle - \tfrac{1}{\sqrt{2}}|1\rangle$$

A la inversa, dado un estado $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, con incertidumbre total, la aplicación de la puerta de Hadamard lo transforma en el estado $|0\rangle$, mientras que si partimos del estado $\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$ obtenemos $|1\rangle$. Por lo tanto, la puerta de Hadamard es inversa de sí misma y permite establecer transiciones entre estados de certidumbre total y de incertidumbre total en el *qubit*. Esta característica resultará de enorme utilidad en el diseño de algoritmos cuánticos. Los estados entrelazados $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ y $\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$ constituyen también una base ortonormal del sistema de un *qubit* y se los representa a menudo de manera abreviada como $|+\rangle$ y $|-\rangle$, respectivamente. Así pues, podemos considerar la transformación de Hadamard como un cambio de

sistema de referencia entre las bases ortonormales $\{|0\rangle, |1\rangle\}$ y $\{|+\rangle, |-\rangle\}$.

Pasemos ahora a ver cómo serán las puertas lógicas en sistemas de múltiples *qubits*. Siguiendo el esquema que hemos aplicado al caso de un *qubit*, podríamos intentar buscar transformaciones análogas a las puertas clásicas de dos bits, tales como las AND, OR y XOR. Pero aquí surge un problema importante. Al definir las puertas lógicas cuánticas como endomorfismos unitarios, hemos introducido una diferencia esencial con el caso clásico: la reversibilidad. Un endomorfismo unitario es siempre invertible, por lo que las puertas lógicas sobre n *qubits* de entrada tendrán inevitablemente como salida también n *qubits* y podremos recuperar los valores de entrada conociendo los valores de salida. Esta reversibilidad de las puertas lógicas cuánticas tiene implicaciones importantes. En primer lugar, no podremos encontrar puertas análogas a las AND, OR y XOR que pasen dos *qubits* a un *qubit*, como ocurre con los bits clásicos. La solución está en restringir las analogías a las puertas clásicas reversibles. Una segunda implicación importante de la reversibilidad de las puertas lógicas cuánticas es que su operación no produce aumento de entropía (es decir, no se pierde información), por lo que la computación cuántica no está sujeta al principio de Landauer, que impone límites termodinámicos al procesamiento de información mediante puertas lógicas irreversibles [21, p. 153].

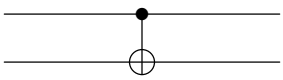
Una vez que hemos restringido la búsqueda de equivalentes cuánticos a aquellas puertas lógicas clásicas que son reversibles, ¿qué otras puertas clásicas se podrán traducir al lenguaje cuántico? En los sistemas de dos bits existe una puerta lógica reversible, que es la puerta NOT controlada (*controlled NOT gate*, en inglés) o CNOT, que deja los dos bits inalterados si el primer bit es 0 y aplica una puerta NOT al segundo bit si el primero vale 0. Se trata evidentemente de una puerta reversible en que los dos bits se comportan de manera totalmente simétrica. El valor resultante del segundo bit es igual al resultado de la puerta XOR, por lo que en el mundo clásico esta puerta CNOT no es más que una puerta XOR a la que se le añade como salida adicional el estado de uno de los bits iniciales para hacerla reversible. Esto no parece aportar gran cosa en el ámbito de los circuitos lógicos clásicos, pero su equivalente cuántico nos va a resultar muy útil en la computación cuántica.

La versión cuántica de la puerta lógica CNOT tendrá que respetar el comportamiento que acabamos de describir para la puerta CNOT clásica, por lo que los estados $|00\rangle$ y $|01\rangle$ deberían permanecer invariables, mientras que los $|10\rangle$ y $|11\rangle$ se transformarán el uno en el otro. La matriz coordenada de esta puerta lógica será entonces la siguiente:

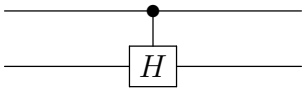
$$CNOT = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{3.10}$$

La representación esquemática de la puerta CNOT en circuitos cuánticos es la

siguiente:



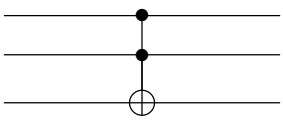
Este concepto de control en dos *qubits* puede extenderse a cualquier puerta lógica sobre un *qubit*, de modo que la puerta lógica se aplique solamente si el *qubit* de control tiene el valor 1. La representación gráfica de estas puertas controladas utiliza también un punto grueso del que surge una línea, como en el caso de la puerta CNOT. Por ejemplo, una puerta de Hadamard sobre el segundo *qubit* controlada por el primer *qubit* se representa de la siguiente manera:



En los sistemas de tres *qubits* existe una puerta lógica similar a la puerta CNOT que acabamos de ver en dos *qubits*. Es la puerta CCNOT o puerta de Toffoli. En la versión clásica de esta puerta hay dos bits de control y al tercer bit se le aplica una inversión solamente cuando los dos primeros bits valen 1. En los demás casos, el tercer bit permanece inalterado. La operación es evidentemente la inversa de sí misma y, por tanto, reversible al igual que la puerta CNOT, por lo que podemos construir una puerta cuántica de Toffoli [9, p. 10]. En la versión cuántica, los vectores base $|110\rangle$ y $|111\rangle$ se transformarán entre sí, mientras que los seis vectores base restantes permanecerán inalterados. La matriz de esta puerta lógica es entonces la siguiente:

$$CCNOT = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \tag{3.11}$$

La representación esquemática de la puerta de Toffoli en circuitos cuánticos es la siguiente:



La puerta de Toffoli clásica tiene una propiedad extraordinaria: se trata de una puerta universal. Esto quiere decir que cualquier otra puerta lógica, como las AND,

NAND, OR, XOR no reversibles que hemos mencionado, puede ser escrita como una combinación de puertas de Toffoli. Dado que podemos replicar el comportamiento de la puerta de Toffoli clásica mediante una versión cuántica, llegamos a la conclusión de que ¡podemos escribir cualquier circuito lógico clásico en forma cuántica! [9, p. 10] Esto significa que la computación clásica es un subconjunto de la computación cuántica, un resultado tan importante que le daremos rango de teorema:

Teorema 3.1. *Todo circuito lógico clásico puede reproducirse como circuito lógico cuántico.*

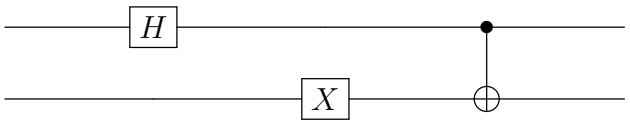
Demostración. Dado un circuito lógico clásico, se puede escribir en función de la puerta universal de Toffoli, que admite una versión cuántica. □

Hasta aquí hemos introducido puertas lógicas cuánticas para sistemas de uno, dos y tres *qubits*. Dado que los sistemas de *qubits* se construyen como productos tensoriales desde los sistemas más simples de un *qubit*, las puertas de un sistema de n *qubits* pueden aplicarse a cualquier subespacio de $m > n$ *qubits*. Podemos dar una definición formal para esta operación:

Definición 3.3. Dada una puerta lógica U sobre un sistema de n *qubits* H_n, llamamos **aplicar la puerta lógica** U **a** n ***qubits*** de un sistema de m *qubits* H_m con $m > n$ a la aplicación del endomorfismo unitario producto tensorial de la puerta lógica U sobre un subespacio de dimensión 2^n de H_m con la aplicación identidad sobre el sistema complementario de dimensión 2^{m-n}.

En representación matricial, el producto tensorial de endomorfismos unitarios se manifiesta con la operación matricial llamada producto de Kronecker, en la que cada entrada de la primera matriz multiplica a una caja con toda la segunda matriz. La mejor manera de entender cómo funcionan estos productos tensoriales es aplicarlo a un ejemplo sencillo.

Sea un circuito como el que se muestra a continuación:



(3.12)

En primer lugar tenemos una puerta de Hadamard sobre el subsistema del primer *qubit*. Si solo existiera ese *qubit*, la operación consistiría en la matriz 2×2 que conocemos, pero al estar en un sistema de dos *qubits*, espacio de Hilbert de dimensión 4, necesitaremos una matriz 4×4, que obtenemos haciendo el producto de Kronecker de la matriz de Hadamard por la identidad:

$$U_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \otimes \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{pmatrix} \tag{3.13}$$

Esta será entonces la matriz correspondiente al primer endomorfismo unitario del circuito. Después nos encontramos con una puerta X sobre el segundo *qubit*, que corresponderá a un producto de Kronecker de la identidad por la matriz X 2×2:

$$U_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \otimes \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{3.14}$$

En tercer lugar, tenemos una puerta CNOT que abarca los dos *qubits* del sistema, por lo que tendra la forma 4×4 ya conocida:

$$U_3 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{3.15}$$

Ahora podemos multiplicar las tres matrices (3.13), (3.14) y (3.15) para obtener la matriz del endomorfismo unitario del circuito completo. Nótese que las trasformaciones ocurren de izquierda a derecha, por lo que, asumiendo el convenio en que los vectores se representan como matrices columna multiplicadas por la derecha, las matrices deben multiplicarse de derecha a izquierda. La matriz de este circuito sencillo de ejemplo será entonces:

$$U = U_3 U_2 U_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{pmatrix} \tag{3.16}$$

Esta manera de manipular las matrices mediante productos de Kronecker es muy importante para construir circuitos cuánticos, por lo que es conveniente adquirir una buena familiarización con su funcionamiento. Nótese que en este circuito de ejemplo (3.12) hemos escrito las puertas de un *qubit* separadas pero podríamos haberlas escrito en paralelo, una sobre la otra, ya que al afectar a subsistemas disjuntos se puede demostrar fácilmente que conmutan entre sí. De hecho, en este caso el producto $U_1 U_2$ lo podríamos haber obtenido directamente como producto de Kronecker de H y X. En este sentido, la propia topología del circuito indica las operaciones equivalentes.

Una vez que nos hemos familiarizado con el funcionamiento de los circuitos lógicos cuánticos, concluiremos esta introducción con una cuestión muy importante: ¿existen puertas lógicas cuánticas universales?

Esta última pregunta que acabamos de formular surge de manera natural al introducir las puertas lógicas cuánticas, pues en el caso clásico sí existen puertas universales, tales como las puertas de Toffoli, NAND y NOR, con las que se pueden construir todas las demás. Pero la respuesta en el caso cuántico es negativa. Y es más, ni siquiera es posible determinar un número finito o numerable de puertas lógicas que tengan ese carácter universal. La razón estriba en la existencia de puertas como R_θ que dependen de una fase θ, que es un número real con un espectro continuo de valores posibles. No obstante, la situación no es tan mala como podría parecer ya que se puede demostrar que cualquier puerta cuántica lógica puede aproximarse tanto como se quiera mediante el uso de un número finito de puertas. De hecho, utilizando tres *qubits* existe una familia de puertas lógicas, las puertas de Deutsch, que constituyen una generalización de la puerta de Toffoli. Dado que se trata de una transformación unitaria sobre un sistema compuesto por tres *qubits*, la puerta de Deutsch será una matriz 8×8. Su forma es la siguiente:

$$D(\alpha) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & i\cos\alpha & \sin\alpha \\ 0 & 0 & 0 & 0 & 0 & 0 & \sin\alpha & i\cos\alpha \end{pmatrix} \tag{3.17}$$

La dependencia del parámetro α indica que se trata de una familia de puertas lógicas (que incluye como caso particular a la puerta de Toffoli, cuando $\alpha = \frac{\pi}{2}$), pero fijando α como un múltiplo irracional de π, cualquier valor de ángulo podrá aproximarse hasta una precisión arbitraria mediante la aplicación reiterada de $D(\alpha)$, habida cuenta que se cumple la propiedad $D(\alpha)D(\alpha') = D(\alpha + \alpha')$ [19] .

Un conjunto de puertas alternativo que también permite aproximar cualquier puerta cuántica es el formado por las puertas H, $R_{\frac{\pi}{4}}$ y $CNOT$. Aunque consta de varias puertas, son todas de uno o dos *qubits*, por lo que su implantación práctica es más sencilla que el de las puertas de Deutsch.

Hemos visto así cómo los principios de superposición y de evolución unitaria nos permiten diseñar puertas lógicas diferentes de las que conocemos en la computación clásica. Con estas puertas lógicas podremos entonces construir algoritmos que serán también distintos de los que permiten las puertas clásicas. A continuación, mencionaremos algunos de los más relevantes y sorprendentes.

4 Los algoritmos cuánticos

Tal como indica el teorema (3.1), cualquier circuito lógico clásico puede ser construido por puertas cuánticas, lo cual resulta bastante natural, pues parece razonable intuir que los fenómenos clásicos sean un caso límite macroscópico de los cuánticos. Despejada la duda de si la computación cuántica incluye a la clásica al menos como caso particular, la pregunta más interesante que se nos plantea es si la computación cuántica puede ir más allá de la clásica en sus capacidades algorítmicas. Por un lado, la superposición de estados parece apuntar a que los *qubits* serían elementos de información mucho más ricos que los bits clásicos, pero no está tan claro que sea así debido al principio de observación que implica que, por mucho que el *qubit* en ausencia de mediciones pueda adoptar todo un continuo de valores en un espacio vectorial complejo de dimensión 2, en el momento en que es observado el valor resultante solamente puede ser uno de los dos valores clásicos 0 y 1. El resultado sorprendente es que la riqueza de estados del *qubit* en ausencia de medición sí se manifiesta en capacidades algorítmicas superiores a las de los algoritmos clásicos.

4.1 El algoritmo de Deutsch

El primer ejemplo histórico de algoritmo que aprovecha estas capacidades de cálculo tan especiales de las puertas lógicas cuánticas es debido a David Deutsch [7] y se remonta a 1985. El problema que resuelve el algoritmo de Deutsch es en gran medida artificial, pero de enorme relevancia académica, ya que muestra el tipo de situación en que los algoritmos cuánticos pueden hacer en una sola operación lo que clásicamente requeriría intentos múltiples. La superposición de estados proporciona así una forma de paralelismo computacional en que las operaciones sobre *qubits* pueden esconder toda una familia parametrizada de operaciones. Además de su interés histórico, el algoritmo de Deutsch es muy ilustrativo de cómo funcionan los algoritmos cuánticos, por lo que merece la pena detenerse a entenderlo en detalle.

El problema que se planteó Deutsch es el siguiente. Supongamos que tenemos una función definida en un bit $f : \{0, 1\} \to \{0, 1\}$. Existen entonces cuatro posibles definiciones para f:

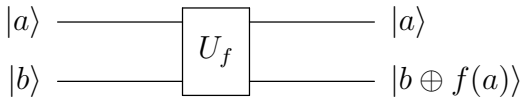
$$\begin{aligned} f_1(0) &= 0; f_1(1) = 0 \\ f_2(0) &= 1; f_2(1) = 1 \\ f_3(0) &= 0; f_3(1) = 1 \\ f_4(0) &= 1; f_4(1) = 0 \end{aligned} \tag{4.1}$$

Suponemos que f es una especie de caja negra y que solamente podemos saber de cuál función se trata aplicándola a los valores 0 y 1 y observando el resultado. Si se nos pregunta entonces a cuál de las cuatro funciones posibles f_1, f_2, f_3 o f_4 corresponde la f de la que disponemos, tendremos que hacer dos evaluaciones de la

función. Solamente calculando tanto $f(0)$ como $f(1)$ sabremos con certeza de cuál se trata. El problema de Deutsch es más sutil. En lugar de preguntar de cuál de las cuatro funciones posibles se trata, la pregunta que se formula es simplemente si f es o bien una de las funciones f_1 o f_2 o bien una de las funciones f_3 o f_4; es decir, si se trata de una función constante o de una función suprayectiva que arroja valores diferentes para cada entrada. Lo interesante del problema radica en que la respuesta que se nos pide requiere menos información que si se nos preguntara de cuál de las cuatro funciones en concreto se trata. Pero aun cuando se nos pide una respuesta con menor detalle, la estrategia clásica para resolver el problema es la misma: habrá que observar el resultado de hacer pasar 0 y 1 por la caja negra. Si al calcular $f(0)$ obtenemos 0 podría tratarse tanto de f_1 como de f_3, mientras que si obtenemos 1, entonces podría ser tanto f_2 como f_4. Solamente evaluando también $f(1)$ sabremos si se trata de una función constante o no y, de hecho, sabremos exactamente cuál de las cuatro funciones es f.

Parecería que la pregunta de si la caja negra representa una función constante o no es inseparable de la pregunta de cuál de las posibles funciones tenemos ante nosotros, pues necesitaremos dos consultas de la función f para resolver el problema. Sorprendentemente, utilizando un circuito cuántico de dos *qubits* se puede responder a la pregunta de si se trata de una función constante (f_1 o f_2) o no (f_3 o f_4) con una sola evaluación de la función. El algoritmo de Deutsch es precisamente este circuito cuántico.

Para escribir el circuito lógico, tenemos primero que expresar la función f como un endomorfismo unitario. Si expresamos las funciones f_1, f_2, f_3 y f_4 en forma matricial, las matrices resultantes no son unitarias, por lo que hace falta un planteamiento un poco más sofisticado. La solución es una idea feliz, que consiste en traducir la función f a una puerta lógica de dos *qubits* equivalente, de la manera siguiente:

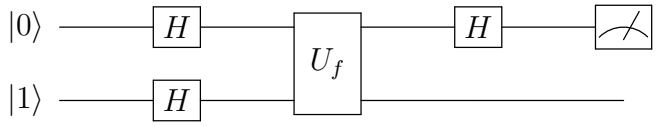


Al tratarse de una puerta lógica sobre dos *qubits*, fijada la base $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$, la representación matricial consistirá en una matriz unitaria 4×4. Asignando a los *qubits* de entrada las cuatro combinaciones de valores posibles y viendo los resultados que se producen con cada una de las cuatro funciones posibles f_1, f_2, f_3 y f_4, podemos construir las cuatro matrices para las puertas lógicas resultantes para cada una de las funciones, que son las siguientes:

$$U_{f_1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad U_{f_2} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$
$$U_{f_3} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \qquad U_{f_4} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{4.2}$$

Con estas cuatro matrices hemos conseguido reescribir las cuatro posibles funciones f de un bit como cuatro transformaciones unitarias que podemos integrar en un circuito lógico cuántico. El problema ahora consiste en que si se nos da una caja negra U_f, que sabemos que ha de ser una de las cuatro matrices anteriores, ¿cómo podemos deducir si se trata o bien de una de las U_{f_1} o U_{f_2} o bien de una de las U_{f_3} o U_{f_4}?

El circuito lógico cuántico que encontró David Deutsch para resolver este problema es el siguiente:



(4.3)

El último elemento del circuito que afecta al *qubit* superior representa la operación de medición.

Vamos a comprobar explícitamente que el circuito anterior (4.3) nos da una solución al problema. Para ello calcularemos su matriz para las cuatro posibles aplicaciones unitarias U_f.

Las dos puertas de Hadamard iniciales tendrán como matriz el producto de Kronecker de las matrices de Hadamard 2×2:

$$H \otimes H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \otimes \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix} \tag{4.4}$$

Por otro lado, la aplicación de una puerta de Hadamard solamente al primer *qubit* como último paso del circuito corresponderá a un producto de Kronecker de la matriz de Hadamard por la identidad:

$$H \otimes I = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \otimes \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{pmatrix} \tag{4.5}$$

La matriz del circuito (4.3), a la que llamaremos D_f será entonces el producto de (4.5) por (4.2) y por (4.4):

$$D_f = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{pmatrix} (U_f) \frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix} \tag{4.6}$$

Sustituyendo los cuatro valores posibles de U_f, (4.2), obtenemos las cuatro formas matriciales posibles para el circuito:

$$\begin{aligned} D_{f_1} &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & -1 \end{pmatrix} \qquad D_{f_2} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 1 & 1 \end{pmatrix} \\ D_{f_3} &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & -1 \\ 0 & 1 & 1 & 0 \\ 0 & -1 & 1 & 0 \end{pmatrix} \qquad D_{f_4} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & -1 \\ 1 & 0 & 0 & 1 \\ 0 & -1 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix} \end{aligned} \tag{4.7}$$

Así pues, al montar el circuito lógico de Deutsch, nuestra caja negra U_f da lugar a una de estas cuatro transformaciones unitarias. El algoritmo descubierto por Deutsch establece además que el estado preparado inicialmente ha de ser el $|01\rangle$. Si hacemos el cálculo de la transformación de este vector mediante cada una de las cuatro D_f posibles, se obtienen los valores siguientes:

$$\begin{aligned} D_{f_1}|01\rangle &= \frac{1}{\sqrt{2}}(|00\rangle - |01\rangle) \\ D_{f_2}|01\rangle &= \frac{1}{\sqrt{2}}(-|00\rangle + |01\rangle) \\ D_{f_3}|01\rangle &= \frac{1}{\sqrt{2}}(|10\rangle - |11\rangle) \\ D_{f_4}|01\rangle &= \frac{1}{\sqrt{2}}(-|10\rangle + |11\rangle) \end{aligned} \tag{4.8}$$

La inspección de los cuatro resultados revela que en el caso de las funciones constantes f_1 y f_2, el vector de estado resultante solamente tiene componentes en los

vectores $\{|00\rangle, |01\rangle\}$ de la base, por lo que al efectuar la operación de medición sobre el primer *qubit*, se observará el estado $|0\rangle$ con total seguridad. En el caso de las funciones f_3 y f_4, en cambio, el estado resultante está en el subespacio engendrado por $\{|10\rangle, |11\rangle\}$, en el que la medición del primer *qubit* da lugar al estado $|1\rangle$. De esta manera, hemos logrado un circuito cuántico que responde a la pregunta de Deutsch con una sola evaluación de la función.

En 1992, David Deutsch y Richard Jozsa publicaron un artículo [8] en el que se presentaba una versión del algoritmo ampliada a un sistema de n *qubits*, en el que se trata de dilucidar si la función de la caja negra es constante o, como posibilidad alternativa, si se trata de una función balanceada que tiene por imágenes 0 y 1 para dos subconjuntos de igual tamaño de los valores de entrada posibles. A esta versión generalizada se le llama algoritmo de Deutsch-Jozsa.

Con posterioridad al algoritmo de Deutsch-Jozsa, se han encontrado otros algoritmos que aceleran de manera notable las computaciones clásicas. Los más interesantes son sin duda los algoritmos de Shor y Grover, ya que reducen la complejidad de operaciones de factorización y búsquedas. Esto tiene repercusiones importantísimas sobre muchas de las tecnologías actuales y, en particular, sobre la seguridad de las primitivas criptográficas de uso habitual. A continuación, expondremos los detalles de estos dos algoritmos.

4.2 El algoritmo de Shor

El algoritmo de Shor fue publicado en el año 1994 [23]. Su autor, Peter Shor, demostró que el problema de la descomposición de un número de n bits en factores primos podía ser resuelto mediante este algoritmo cuántico con complejidad polinomial en el logaritmo del número de bits, concretamente $O((\log n)^3)$. Esto contrasta con la complejidad exponencial que presenta el problema en cualquier enfoque clásico y resulta demoledor para algunas de las técnicas criptográficas de uso actual. A continuación daremos una breve descripción del algoritmo.

Como paso previo para formular el algoritmo de Shor, necesitaremos un algoritmo cuántico muy importante: la transformada discreta de Fourier. Utilizando este algoritmo cuántico muy versátil se definen otros dos algoritmos: el de búsqueda de períodos y el de búsqueda de orden modular. Este último constituye la parte esencial del algoritmo de factorización de Shor. Vamos a ver en primer lugar cómo se construyen estos algoritmos auxiliares.

4.2.1 La transformada cuántica de Fourier

La transformada discreta de Fourier es una función que convierte n números complejos $\{x_k\}_{k=1,\ldots,n}$ en otros n números complejos $\{y_k\}_{k=1,\ldots,n}$ mediante la siguiente ley:

$$y_k = \frac{1}{\sqrt{n}} \sum_{j=0}^{n-1} e^{\frac{jk}{n} 2\pi i} x_j \tag{4.9}$$

Este tipo de transformación puede considerarse un endomorfismo en un espacio vectorial complejo de dimensión n, por lo que puede definirse de igual manera en el espacio de Hilbert H de un sistema de n *qubits*. En este caso, H tendrá dimensión 2^n y definiremos la transformada discreta de Fourier de manera que las 2^n coordenadas de cualquier vector de H se transformen de acuerdo con (4.9). Para ello, los vectores de la base $\{|i\rangle\}_{i=0,\dots,2^n-1}$ tendrán que transformarse de manera análoga bajo esta aplicación, a la que denominaremos QFT:

$$QFT|j\rangle = \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} e^{\frac{jk}{2^n} 2\pi i} |k\rangle \tag{4.10}$$

Se puede demostrar por desarrollo directo que esta aplicación QFT es unitaria. Vamos a ver la forma que adopta para las dimensiones más bajas.

En un sistema de un solo *qubit*, $n = 1$, las fórmulas de (4.10) son:

$$\begin{aligned} QFT_1|0\rangle &= \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \\ QFT_1|1\rangle &= \frac{1}{\sqrt{2}}(|0\rangle + e^{\pi i}|1\rangle) = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \end{aligned} \tag{4.11}$$

La matriz coordenada en la base $\{|0\rangle, |1\rangle\}$ es entonces:

$$QFT_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = H \tag{4.12}$$

Encontramos así el resultado interesante de que la transformada cuántica de Fourier para un sistema de un *qubit* es precisamente la puerta de Hadamard. Esto explica intuitivamente el hecho de que la transformada cuántica de Fourier sea un componente tan habitual en los algoritmos cuánticos, ya que se puede considerar una generalización de la puerta de Hadamard a sistemas de múltiples *qubits*.

En un sistema de dos *qubits*, $n = 2$, y las fórmulas de (4.10) serán:

$$
\begin{aligned}
QFT_2|00\rangle &= \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle) \\
QFT_2|01\rangle &= \frac{1}{2}(|00\rangle + e^{\frac{\pi i}{2}}|01\rangle + e^{\frac{2\pi i}{2}}|10\rangle + e^{\frac{3\pi i}{2}}|11\rangle) \\
QFT_2|10\rangle &= \frac{1}{2}(|00\rangle + e^{\frac{2\pi i}{2}}|01\rangle + e^{\frac{4\pi i}{2}}|10\rangle + e^{\frac{6\pi i}{2}}|11\rangle) \\
QFT_2|11\rangle &= \frac{1}{2}(|00\rangle + e^{\frac{3\pi i}{2}}|01\rangle + e^{\frac{6\pi i}{2}}|10\rangle + e^{\frac{9\pi i}{2}}|11\rangle)
\end{aligned}
\tag{4.13}
$$

Haciendo el cambio de variable $\omega = e^{\frac{\pi i}{2}}$ y teniendo en cuenta la relación $e^{2\pi i} = 1$, las ecuaciones anteriores adoptan la siguiente forma más compacta:

$$
\begin{aligned}
QFT_2|00\rangle &= \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle) \\
QFT_2|01\rangle &= \frac{1}{2}(|00\rangle + \omega|01\rangle + \omega^2|10\rangle + \omega^3|11\rangle) \\
QFT_2|10\rangle &= \frac{1}{2}(|00\rangle + \omega^2|01\rangle + |10\rangle + \omega^2|11\rangle) \\
QFT_2|11\rangle &= \frac{1}{2}(|00\rangle + \omega^3|01\rangle + \omega^2|10\rangle + \omega|11\rangle)
\end{aligned}
\tag{4.14}
$$

La matriz coordenada en la base $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$ es entonces:

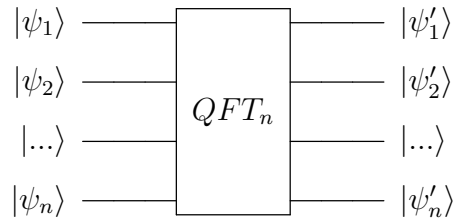
$$
QFT_2 = \frac{1}{2}\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & \omega & \omega^2 & \omega^3 \\ 1 & \omega^2 & 1 & \omega^2 \\ 1 & \omega^3 & \omega^2 & \omega \end{pmatrix} \tag{4.15}
$$

Haciendo el cálculo análogo para el sistema de tres *qubits*, $n = 3$, y adoptando en este caso el cambio de variable $\omega = e^{\frac{\pi i}{4}}$, la matriz que se obtiene es la siguiente:

$$
QFT_3 = \frac{1}{\sqrt{8}}\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & \omega & \omega^2 & \omega^3 & \omega^4 & \omega^5 & \omega^6 & \omega^7 \\ 1 & \omega^2 & \omega^4 & \omega^6 & 1 & \omega^2 & \omega^4 & \omega^6 \\ 1 & \omega^3 & \omega^6 & \omega & \omega^4 & \omega^7 & \omega^2 & \omega^5 \\ 1 & \omega^4 & 1 & \omega^4 & 1 & \omega^4 & 1 & \omega^4 \\ 1 & \omega^5 & \omega^2 & \omega^7 & \omega^4 & \omega & \omega^6 & \omega^3 \\ 1 & \omega^6 & \omega^4 & \omega^2 & 1 & \omega^6 & \omega^4 & \omega^2 \\ 1 & \omega^7 & \omega^6 & \omega^5 & \omega^4 & \omega^3 & \omega^2 & \omega \end{pmatrix} \tag{4.16}
$$

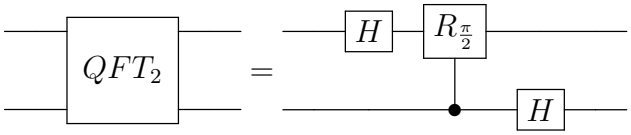
Mediante este tipo de desarrollos tediosos de la fórmula general podemos construir la matriz unitaria para el sistema de n *qubits* con cualquier n. En principio, podríamos conformarnos con esto y aceptar esas formas explícitas como definición de la puerta lógica QFT_n para cada valor de n. En los circuitos, representaremos a

la transformada de Fourier mediante una caja con las letras QFT y el subíndice del número de *qubits* al que afecta:



(4.17)

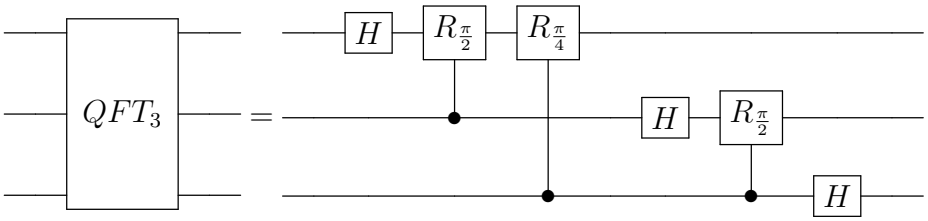
La puerta lógica QFT_n puede expresarse también de manera inductiva a partir de QFT_{n-1}. Observemos primero que QFT_2 puede representarse mediante el siguiente circuito:



(4.18)

Puede comprobarse fácilmente esta relación entre circuitos calculando la matriz del de la derecha mediante productos de Kronecker y productos matriciales ordinarios (como se explicó en una sección precedente) y viendo que coincide con la matriz (4.15) que habíamos visto para QFT_2.

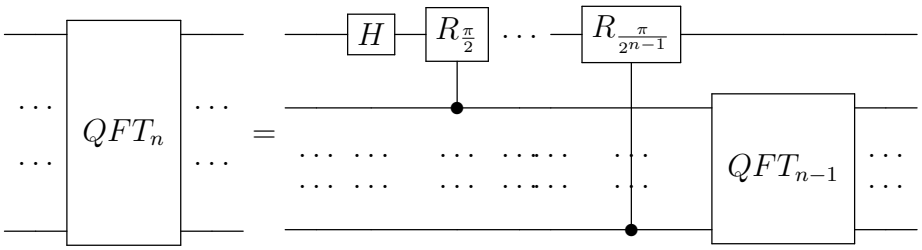
Para la puerta QFT_3 se puede construir un circuito análogo, que es el siguiente:



(4.19)

Como en el caso anterior, la igualdad de los dos circuitos se puede compobar de manera rutinaria calculando la matriz del circuito de la derecha, que coincide con (4.16).

Comparando los circuitos (4.18) y (4.19) se intuye la forma general que permite una definición de tipo inductivo, que sería la siguiente:



(4.20)

La demostración de esta forma general se basa en desarrollos algebraicos a partir de la definición de la transformada de Fourier (4.10) y puede encontrarse en el libro de Nielsen y Chuang [21, p. 218]. Gracias a esta definición inductiva, si se dispone de la tecnología para construir circuitos cuánticos con puertas de Hadamard H y puertas de fase R_θ, se podrán construir de manera trivial las puertas QFT para diferentes valores de n.

4.2.2 La búsqueda de períodos

La transformada cuántica de Fourier QFT (junto a su inversa $QFT^\dagger$) que acabamos de introducir es un componente importante en muchos algoritmos cuánticos. En particular, una de sus aplicaciones más útiles consiste en encontrar el período de una función periódica. El problema que se plantea en este caso es el siguiente: sea una función f sobre un conjunto de n bits que devuelve n bits. Sea la función f periódica, de tal modo que considerando las combinaciones de valores de los n bits como el intervalo de los números naturales entre 0 y 2^n-1, se cumpla $f(x+r) = f(x)$ para un cierto valor r, tomando la suma módulo 2^n para evitar salir del intervalo de definición. Se impone, además, que la función f no repita valores dentro de un mismo período; es decir, que $f(x) = f(y)$ si y solo si $y \equiv x \mod r$. El problema consiste en que se nos da la función f como una caja negra, tal como ocurría en el problema del algoritmo de Deutsch, y se nos pide que determinemos el período r.

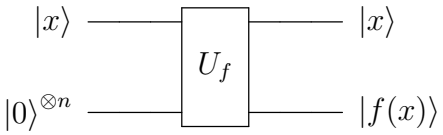
En computación clásica, el problema tal como lo hemos planteado requeriría a lo sumo r evaluaciones de la función f en la solución más ingenua, evaluando $f(0)$, $f(1)$, ... hasta encontrar el valor r para el que $f(r) = f(0)$ (aquí resulta necesaria la condición de que f sea inyectiva dentro de un período). El comportamiento de este algoritmo sería de tipo $O(r)$, lineal en r, pero exponencial, $O(2^n)$, respecto al número de bits n. En realidad, hay algoritmos clásicos que mejoran el rendimiento del algoritmo ingenuo por fuerza bruta, pero el comportamiento es siempre, en cualquier caso, exponencial respecto al número de bits.

Cuando pasamos al mundo de la computación cuántica, el problema que acabamos de plantear se puede resolver con comportamiento polinomial $O(n)$ en el número de bits gracias a la transformada cuántica de Fourier. ¿Por qué es la transformada de Fourier útil para resolver este problema? La razón se encuentra en un rasgo de la transformada discreta de Fourier, que es la manera en que transforma un conjunto de

valores periódicos en otro conjunto en el que los únicos valores no nulos se encuentran separados entre sí una distancia igual al valor del período en la preimagen. Es decir, si tenemos un conjunto de valores $\{a_0, a_1, \ldots, a_r, a_0, a_1, \ldots, a_r, \ldots, a_0, a_1, \ldots, a_r\}$ en el que la parte $\{a_0, \ldots, a_r\}$ se repite un número s de veces, entonces su transformada de Fourier será un conjunto de valores $\{b_0, 0, \ldots, 0, b_1, 0, \ldots, 0, \ldots, b_{s-1}, 0, \ldots, 0\}$ en el que los valores no nulos ocupan las posiciones del valor inicial a_0 en los datos no transformados [27]. Esta propiedad es la que explica un uso muy importante de la transformada cuántica de Fourier para detectar períodos en el estado de un sistema de *qubits*.

Siguiendo la exposición de Umesh Vazirani [27], resulta conveniente simplificar el problema asumiendo que el conjunto de datos iniciales contienen un número entero de períodos[2]; es decir, que el periodo r divide al número de valores 2^n. También es necesario que el período r sea suficientemente pequeño para que se manifieste el carácter periódico de la función f. No basta con el caso extremo en que $r = \frac{2^n}{2}$, sino que se precisa que r sea tal que $r < 2^{\frac{n}{2}}$.

Una vez detallada la naturaleza del problema que se pretende resolver, nos falta definir cómo se expresará la función f como endomorfismo unitario que se pueda insertar en un circuito lógico cuántico, de manera análoga a como hicimos en el algoritmo de Deutsch. En este caso, la manera de dar expresión unitaria a una función f viene dada por el siguiente circuito de $2n$ *qubits*:

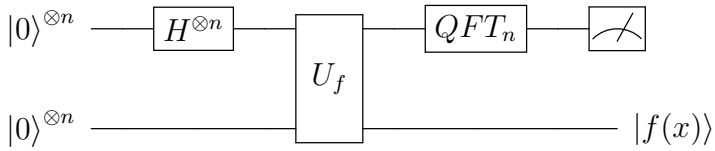


(4.21)

Para ejecutar el circuito y poder conseguir nuestro objetivo, necesitaremos en primer lugar que el estado de entrada de los primeros n *qubits*, representados como $|x\rangle$ en el circuito, esté completamente entrelazado, lo cual puede obtenerse de la manera habitual aplicando puertas de Hadamard a un estado preparado inicialmente como $|0\rangle^{\otimes n}$[3]. Si tras pasar por U_f sometemos a los n *qubits* de la parte inferior del circuito, $|f(x)\rangle$, a una medición, obtendremos aleatoriamente una de las posibles imágenes de f. Pero sabemos que el valor obtenido x_0 se repite con período r, por lo que el estado de los n *qubits* superiores será una superposición a la que solamente contribuyen valores $x_0, x_0 + r, x_0 + 2r, ...$ aislados únicos en cada período, aquellos para los que f da el valor medido. Es a ese estado de superposición al que se ha de aplicar la transformada de Fourier. Al hacerlo, las propiedades de la transformada de Fourier implican que se obtendrá un estado tal que al medirlo arrojará un valor múltiplo del período r. Estas ideas nos llevan al siguiente circuito:

[2]Como se menciona más adelante, esta simplificación no es válida para el algoritmo completo, pero elimina una parte difícil que se resuelve con un algoritmo clásico, por lo que la complicación del caso general no está relacionada con la parte cuántica que nos interesa.

[3]A veces se utiliza la transformación QFT_n en lugar de $H^{\otimes n}$; el efecto es el mismo.



(4.22)

Dado que ese resultado puede ser cualquier kr con k entero positivo; para determinar el período, el algoritmo consistirá en ejecutar este circuito al menos dos veces hasta obtener dos valores kr y $k'r$ diferentes para después calcular el máximo común divisor de estos dos valores, que será el período buscado. El algoritmo, paso a paso, es entonces el siguiente:

1. Preparar un sistema de $2n$ *qubits*, todos ellos en el estado $|0\rangle$.

2. Ejecutar el circuito (4.22) y guardar el resultado de la medición como número entero n_1 de $[0, 2^n - 1]$.

3. Ejecutar de nuevo el circuito (4.22) y guardar el resultado n_2.

4. Si $n_1 = n_2$, repetir el paso anterior.

5. Calcular el máximo común divisor de n_1 y n_2 mediante el algoritmo de Euclides.

El algoritmo anterior se complica de manera no trivial en el caso general en que r no tiene por qué dividir a 2^n. En ese caso, habría que añadir un paso en el que se utiliza el algoritmo clásico de la expansión en fracciones continuas [21, p. 230]. En esta introducción simplificada, hemos omitido ese paso adicional, que puede consultarse en las referencias.

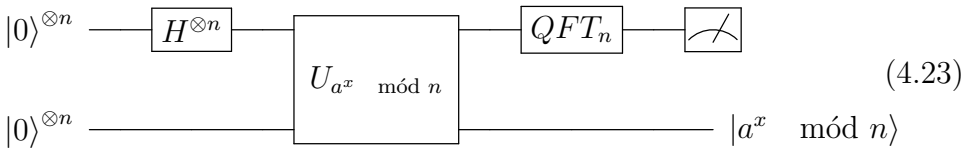
Concluimos esta introducción al algoritmo de búsqueda de período con una observación importante. En la descripción del razonamiento intuitivo que conduce al circuito (4.22), se describió una operación de medición sobre los n *qubits* inferiores del circuito tras pasar por U_f. En el circuito final no aparece esa operación de medición porque, en realidad, no es necesaria. Esos *qubits* no intervienen ya sobre el resultado y su posible medición, aunque ayuda a entender el funcionamiento dle circuito, resulta ser innecesaria. Esto es consecuencia de dos principios importantes que no hemos mencionado y que afectan a la medición en los circuitos cuánticos: el principio de la medición diferida y el principio de la medición implícita [21, p. 187].

4.2.3 El algoritmo de búsqueda de orden modular

Ahora introduciremos otro algoritmo auxiliar que, en realidad, es una aplicación inmediata del anterior. En el siguiente apartado se entenderá por qué este algoritmo es necesario para factorizar números. De momento, describiremos simplemente el problema que se quiere resolver y cómo lograrlo a partir del algoritmo de búsqueda de períodos.

Sean dos números enteros a y n tales que $a < n$ y que no tienen factores comunes; es decir, $mcd(a, n) = 1$. Entonces se llama **orden de** a **módulo** n al mínimo entero $r > 1$ tal que $a^r \equiv 1 \mod n$. Por ejemplo, el orden de 3 módulo 11 es 5, pues se cumple $3^5 \equiv 1 \mod 11$.

Pero en el mundo cuántico, en cambio, disponemos del algoritmo adecuado para resolver el problema en tiempo polinomial, que es precisamente el algoritmo de búsqueda de períodos que vimos en el apartado anterior. Para entender por qué este problema es un caso particular de búsqueda de períodos, basta con observar que si $a^r \equiv 1 \mod n$, entonces $a^{r+1} \equiv a^r a \equiv a \mod n$, $a^{r+2} \equiv a^2 \mod n$, etc. Evidentemente, los valores de las potencias se repiten en ciclos desde a^0 hasta a^{r-1} y a^r marca el comienzo del segundo período. La función $a^r \mod n$ es, por lo tanto, una función periódica con las características exigidas por el algoritmo de búsqueda de períodos. El circuito buscado es entonces simplemente el de (4.22) particularizado a la potencia modular:



(4.23)

4.2.4 El paso final: la búsqueda de factores

Estamos ya en condiciones de resolver el problema de la factorización. El problema consiste simplemente en dado un número natural n, encontrar dos números naturales p y q tales que $n = pq$. Si descubrimos un algoritmo que encuentre esta descomposición, bastará con aplicarlo reiteradamente para descomponer cualquier número en factores primos. Como hemos adelantado, este problema de la factorización se resuelve en computación clásica mediante algoritmos de complejidad no polinomial. Pero en computación cuántica se consigue, como vamos a ver, que el comportamiento pase a ser de tipo $O((\log n)^3)$, siendo $\log n$ el número de bits con el que se puede representar el número n.

El circuito cuántico con el que acabamos el apartado anterior es la pieza clave del rompecabezas del algoritmo de factorización. De hecho, el resto del algoritmo es puramente clásico. Para entender cómo utilizar el algoritmo anterior para la factorización nos hace falta una última idea feliz, que describimos a continuación.

Supongamos que conocemos dos números a y r tales que $a^r \equiv 1 \mod n$ y supongamos que, además, r es par. Entonces $r/2$ es un número natural y se cumple $(a^{\frac{r}{2}})^2 \equiv 1 \mod n$. Desarrollando esta igualdad, se tiene:

$$\begin{aligned} (a^{\frac{r}{2}})^2 &\equiv 1 \mod n \\ (a^{\frac{r}{2}})^2 - 1 &\equiv 0 \mod n \\ (a^{\frac{r}{2}} - 1)(a^{\frac{r}{2}} + 1) &\equiv 0 \mod n \end{aligned} \tag{4.24}$$

Esta última expresión indica que n divide al producto de los dos números $(a^{\frac{r}{2}} - 1)(a^{\frac{r}{2}} + 1)$, por lo que calculando los máximos comunes divisores de n y $(a^{\frac{r}{2}} - 1)$ y de n y $(a^{\frac{r}{2}} + 1)$ mediante el algoritmo de Euclides, tendríamos dos factores de n. La dificultad aquí está en encontrar ese valor a^r y eso es lo que podemos plantear como un problema de búsqueda de orden modular a partir de un a cualquiera. Gracias a esta idea, tenemos ya todas las piezas del rompecabezas para formular el algoritmo definitivo:

1. Elegir un número natural aleatorio $a < n$.

2. Calcular, mediante el algoritmo de Euclides, el máximo común divisor d de a y n.

3. Si $d > 1$, se completa el algoritmo y d y n/d son los factores buscados. Si $d = 1$, se pasa al siguiente paso.

4. Ejecutar el circuito cuántico (4.23) de búsqueda de orden modular para obtener el orden r.

5. Si r es impar volver al primer paso. Si es par, pasar al paso siguiente.

6. Si r es par, calcular mediante el algoritmo de Euclides el máximo común divisor de n con $a^{\frac{r}{2}} - 1$ y con $a^{\frac{r}{2}} + 1$. El segundo podría coincidir con n; en tal caso, volver al primer paso. Si no, los dos valores obtenidos son los factores.

En el quinto paso, podríamos temer que fueran necesarios muchos intentos, pero se puede demostrar que la probabilidad de que el r encontrado sea par es por lo menos $1/2$ [27], por lo que el número de intentos necesarios es normalmente pequeño. El gran cuello de botella en la computación clásica es el cuarto paso y ahí es donde interviene la capacidad del circuito cuántico.

4.3 El algoritmo de Grover

Tras el algoritmo de Shor, el segundo gran algoritmo de la computación cuántica es el algoritmo de Grover. Este algoritmo fue publicado por Lov Grover en 1996 [15] y proporciona una mejora espectacular en el comportamiento de las búsquedas en conjuntos no ordenados de elementos. Si, por ejemplo, disponemos de una lista de N palabras en la que sabemos que hay una aparición de una palabra a, el procedimiento clásico para encontrar a en la lista consiste en una búsqueda lineal, en la que se van cotejando los elementos del conjunto uno a uno con el elemento buscado hasta

encontrarlo. Este procedimiento puede requerir desde una única búsqueda (caso afortunado en el que el primer elemento consultado resulta ser el buscado) hasta N búsquedas (caso de infortunio máximo en el que el elemento buscado es el último comprobado). En general, se necesitará consultar aproximadamente la mitad de la lista de palabras. En términos de complejidad, el algoritmo de búsqueda lineal tiene un comportamiento de tipo $O(N)$. Tanto el sentido común como la teoría dejan claro que no hay ninguna manera más óptima de hacer este tipo de búsqueda en la computación clásica. En computación cuántica, en cambio, el algoritmo de Grover permite realizar este tipo de búsqueda con complejidad $O(\sqrt{N})$. Si los conjuntos en los que se hace la búsqueda son valores representables por n bits, entonces la complejidad del algoritmo de Grover sería $O(2^{\frac{n}{2}})$ frente a $O(2^n)$ del procedimiento clásico.

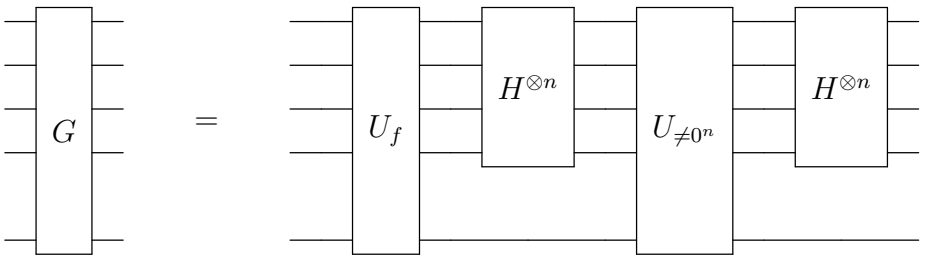
A diferencia de lo que ocurre con el algoritmo de Shor, que convierte en complejidad polinomial lo que clásicamente era complejidad exponencial, la mejora que proporciona el algoritmo cuántico en este caso no es de tipo cualitativo, sino cuantitativo. Aun así, las consecuencias son espectaculares. Por ejemplo, si queremos buscar un valor concreto en un conjunto que pueda tener todos los valores representables en 256 bits, la complejidad de la búsqueda lineal clásica sería $O(2^{256})$ mientras que con el algoritmo de Grover se reduce a $O(2^{128})$; es decir, ¡2^{128} veces más rápido! O lo que es lo mismo, buscar un valor por fuerza bruta en un conjunto de 256 bits mediante el algoritmo de Grover tiene la misma complejidad que el problema análogo en 128 bits mediante la búsqueda lineal clásica.

Una vez introducidas las características del algoritmo, vamos a ver brevemente las ideas básicas necesarias para su formulación.

Para formular el problema de la búsqueda de una manera general que pueda integrarse en un circuito cuántico, asumimos que los elementos entre los que hacemos la búsqueda son cadenas de n bits y que disponemos de una función $f : \{0,1\}^n \to \{0,1\}$ cuyos detalles internos desconocemos, pero que se comporta de modo que $f(x) = 1$ solamente cuando x es la cadena de bits buscada y $f(x) = 0$ para todas las $2^n - 1$ cadenas restantes.

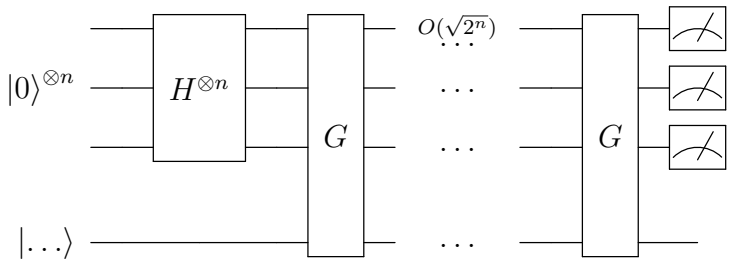
El algoritmo de Grover se basa en dos operaciones que, intuitivamente, van separando el valor x buscado de los demás hasta que su presencia resulta fácilmente detectable. La primera de estas operaciones recibe el nombre de **inversión de fase** y consiste en hacer que el vector de estado $|x\rangle$ que corresponde al elemento x buscado pase a estar multiplicado por un factor i (geométricamente, un giro de 180º en el plano complejo), dejando igual todos los demás estados. Es decir, si el estado del sistema de *qubits* es $|\psi\rangle = \sum_{x\in\{0,1\}^n} \alpha_x|x\rangle$, la inversión de fase lo debe convertir en $|\psi\rangle = \sum_{x\in\{0,1\}^n} \alpha_x(-1)^{f(x)}|x\rangle$. Esta operación se puede expresar como una aplicación unitaria (en una construcción similar a la de la función f del algoritmo de Deutsch, cuyos detalles omitimos), por lo que podemos incorporarla al circuito cuántico como una puerta lógica, a la que llamaremos U_f. La segunda operación que contribuye a esta separación del estado del valor buscado recibe el nombre de **inversión respecto a la media**, y consiste en modificar cada coordenada o amplitud del vector de

estado haciendo que si está por encima del valor medio de todas las coordenadas pase a estar por debajo y viceversa. Matemáticamente, se trata de transformar el estado $\sum_{x\in\{0,1\}^n} \alpha_x|x\rangle$ en $\sum_{x\in\{0,1\}^n}(2\mu-\alpha_x)|x\rangle$, donde $\mu = \frac{1}{2^n}\sum_{x\in\{0,1\}^n}\alpha_x$. Esta transformación también es unitaria, por lo que puede integrarse en un circuito cuántico. Se puede demostrar que esta transformación tiene una forma extremadamente sencilla al pasar de la base de estados observables $\{|0\rangle, |1\rangle\}$ a la base de estados entrelazados $\{|+\rangle, |-\rangle\}$. En esta segunda base ortonormal, la inversión respecto a la media equivale a una puerta lógica como la U_f que ya tenemos, pero para una función g de n bits en un bit tal que $g(0) = 0$ y $g(x) = 1$ si $x \neq 0$. Tendremos entonces una puerta $U_{\neq 0^n}$ aplicada entre dos puertas de Hadamard para n *qubits* que se encargan de la transformación entre las dos bases ortonormales. Aunque no hemos entrado a fondo en los detalles, esta descripción nos permite intuir el aspecto que tendrá el elemento G del circuito cuántico de Grover que realiza estas dos operaciones de inversión de fase e inversión respecto a la media, que se indica en la figura siguiente:



(4.25)

Las puertas U_f y $U_{\neq 0^n}$ utilizan al menos un *qubit* ancilar, necesario para las puertas U_f y $U_{\neq 0^n}$, por lo que abarcan un rango de *qubits* mayor que el de las transformaciones de Hadamard. Finalmente, el algoritmo de Grover consiste en aplicar esta puerta lógica G muchas veces sobre n *qubits* con entrelazamiento máximo, por lo que habrá que preparar n *qubits* en el estado $|0\rangle$ y aplicarles en primer lugar transformaciones de Hadamard para entrelazarlos y, a continuación, aplicarles el circuito (4.25) de manera reiterada. El número de iteraciones ha de ser del orden de $\sqrt{2^n}$. El circuito definitivo del algoritmo de Grover es entonces:



(4.26)

Nótese que a diferencia de algoritmos exactos como el de Deutsch, el de Grover es de tipo probabilístico, al igual que el de Shor. Las inversiones de fase e inversiones respecto a la media podrían con una probabilidad pequeña, pero no nula, producir un falso positivo. Pero al hacer un número grande de iteraciones sobre estas operaciones, la probabilidad de falso positivo tiende a cero y se puede considerar despreciable a todos los efectos.

4.4 Consecuencias de los algoritmos cuánticos para la criptografía

El algoritmo de Grover que acabamos de ver tiene consecuencias importantes para la criptografía de uso actual. Al reducir la complejidad de una búsqueda por fuerza bruta sobre cadenas de n bits a $O(\sqrt{N})$ frente a la complejidad cuántica de $O(N)$, la seguridad de muchas primitivas criptográficas se ve seriamente reducida. Por ejemplo, encontrar por fuerza bruta unos datos cuyo valor SHA-256 (una de las funciones *hash* más populares) coincida con uno dado requeriría un numero de operaciones del orden de $O(2^{256})$ utilizando procedimientos clásicos, pero el algoritmo de Grover permite hacerlo con $O(2^{128})$. Nótese que esta reducción en la complejidad, aun siendo espectacular, no llega a romper la primitiva criptográfica, sino que reduce su seguridad al nivel que tendría una función *hash* equivalente de 128 bits. Para mantener el nivel de seguridad de 256 bits, bastaría con pasar a utilizar funciones *hash* de 512 bits como SHA-512 o la Keccak-512 de la nueva normativa estadounidense SHA-3. Este nivel de seguridad de 256 bits es el que corresponde a ataques de preimagen en que se busca un valor de entrada que dé lugar a un valor **hash** determinado. Cuando se buscan simplemente colisiones, el nivel de seguridad es menor debido al fenómeno conocido como "paradoja del cumpleaños" y las funciones *hash* de 512 bits tienen un nivel de seguridad clásico de 256 bits ($n/2$). En el caso cuántico, existe una versión de la paradoja del cumpleaños basada en una variación del algoritmo de Grover que reduce la seguridad frente a colisiones a $n/3$. Esta versión cuántica de la paradoja del cumpleaños fue descrita originalmente por Gilles Brassard, Peter Høyer y Alain Tapp [6] y hace que las funciones mencionadas *hash* de 512 bits tengan una seguridad cuántica frente a colisiones de $512/3 \approx 170$ bits.

Muchísimo más crítico es el efecto del algoritmo de Shor sobre las primitivas criptográficas de uso actual. Este algoritmo reduce el problema de la factorización a una complejidad polinomial en el logaritmo del número de bits, concretamente $O((\log n)^3)$. Esto contrasta con la complejidad exponencial que presenta el problema en cualquier enfoque clásico y resulta demoledor para algunas de las técnicas criptográficas de uso actual. En particular, la criptografía de clave pública RSA se basa en obtener números primos suficientemente grandes para que sea inviable recuperarlos cuando únicamente se conoce su producto. De esta manera se generan las claves privadas (los factores) y pública (el producto) con las que se firman o cifran mensajes. La seguridad de RSA depende precisamente de la inviabilidad de los algoritmos conocidos de factorización. Pero si se construyeran ordenadores cuánticos capaces de ejecutar el algoritmo de Shor, esa factorización sería posible. El algoritmo RSA quedaría roto y se convertiría en un algoritmo de interés puramente histórico y académico, sin aplicación práctica. Esto obligaría a cambiar muchos sistemas de seguridad informática como los métodos de autenticación segura de sitios web, que se suelen basar en RSA.

También otros sistemas habituales de criptografía de clave pública como DSA (*Digital Signature Algorithm*) y ECDSA (*Elliptic Curve Digital Signature Algorithm*)

son vulnerables a variantes del algoritmo de Shor. Con leves modificaciones, se pueden obtener versiones del algoritmo de Shor que resuelven el problema del logaritmo discreto [24], con lo que los sistemas de clave pública basados en exponenciación son también víctimas de la potencia de este algoritmo. Un área en la que se manifestaría este problema es el de las criptomonedas como Bitcoin, en las que las transacciones monetarias se firman con una clave pública ECDSA. Un atacante que dispusiera de un ordenador cuántico capaz de ejecutar la versión modificada del algoritmo de Shor sería capaz de encontrar claves privadas para claves públicas conocidas utilizadas previamente en la red Bitcoin. Esta es la razón principal por la que existe una recomendación generalizada en los pagos en criptomonedas de no reutilizar direcciones [4] [5]. Las direcciones Bitcoin que se utilizan para recibir pagos se codifican a partir de un valor *hash* de la clave pública, por lo que es imposible conocer la clave pública correspondiente a una dirección que ha recibido un pago. Pero cuando se realiza un pago a partir de la dirección en que se recibió un pago previo, la clave pública se da a conocer en la red para que se pueda verificar la firma ECDSA. Aunque ese problema se elimina con la práctica de no reutilizar direcciones, el desarrollo futuro de ordenadores cuánticos, exigiría reemplazar el uso de ECDSA en Bitcoin y otras criptomonedas por nuevas primitivas criptográficas resistentes a los algoritmos cuánticos.

5 Escollos tecnológicos para la realización de ordenadores cuánticos

Las capacidades computacionales que teóricamente ofrecen los ordenadores cuánticos son sin duda un aliciente para su desarrollo material. Sin embargo, las dificultades técnicas que bloquean su desarrollo son enormes. Basta con recordar que el mayor logro del algoritmo de Shor mencionado en la anterior sección ha sido factorizar el número 21. Tal como suena, el resultado de $21 = 3 \times 7$ es el mayor éxito logrado hasta hoy por la computación cuántica [26]. Esto evidencia que la construcción de un ordenador cuántico comparable con los ordenadores actuales de naturaleza clásica se enfrenta aún a problemas técnicos no resueltos.

Estos problemas técnicos para producir ordenadores cuánticos proceden de dos tipos de dificultades diferentes:

1. **La "decoherencia cuántica"**. El comportamiento antiintuitivo de la mecanica cuántica en que la mera observación de un estado lo modifica proyectándolo sobre la base de autovectores hace sumamente difícil mantener la evolución de un sistema cuántico sin que se produzca esa proyección. Para que las puertas lógicas cuánticas funcionen de acuerdo con la teoría, es necesario mantener el sistema libre de observación, de tal modo que pueda evolucionar unitariamente sin que se produzca el indeseado colapso debido a la observación. Cualquier pequeña perturbación que implique extraer información del estado equivale físicamente a una medición; de ahí la dificultad de mantener el sistema de *qubits* aislado de tales perturbaciones.

2. **La necesidad de códigos correctores**. Los sistemas cuánticos son mucho más susceptibles de errores que los clásicos. Mientras que un bit clásico solamente puede adoptar dos valores discretos, 0 y 1, el estado de un *qubit* puede adoptar un continuo de valores sobre la esfera unitaria del espacio de Hilbert de los estados. Esto hace que, incluso si se logran evitar las perturbaciones provocadas por las mediciones involuntarias, la decoherencia antes mencionada, haya también que evitar que se den alteraciones en las transformaciones unitarias del sistema, que provocarían resultados inesperados.

Este segundo problema de los códigos correctores ha resultado ser mucho más fácil de tratar que el problema de la decoherencia. En los últimos veinte años, se ha desarrollado toda una teoría de códigos correctores muy esperanzadora. En el resto del trabajo, presentaremos los principales resultados de este campo de investigación. Dado que muchos aspectos de la teoría cuántica de códigos correctores encuentran una analogía en la teoría clásica de códigos, haremos primero una breve introducción a esta última. Así, introduciremos en primer lugar algunos conceptos de la teoría de códigos clásicos que son necesarios también en la teoría cuántica.

6 Modelo de errores en computación clásica

Como referencia para el estudio de los errores en computación cuántica, introduciremos primero aquí brevemente la manera en que se modelan los errores en computación clásica.

En un ordenador clásico, la información se almacena en un número finito $n \in \mathbb{N}$ de bits. Un bit no es más que un conjunto finito de dos elementos a los que convencionalmente se suele representar por $\{0, 1\}$. En n bits tenemos entonces una sucesión de valores $\{b_i\}_{i=1,\ldots,n}$ cada uno de los cuales puede ser 0 o 1. En la sección siguiente daremos una definición más precisa de estas sucesiones de bits, a las que llamaremos "cadenas binarias".

Supongamos que se dispone de información codificada en forma de una cadena binaria de n bits (como podría ser, por ejemplo, un número entero representado mediante una cadena de 32 bits). Si esta información debe transmitirse a través de un canal de comunicación en el espacio o simplemente conservarse sin cambios durante el transcurso de un cierto tiempo, podría ser que la información se viera alterada como resultado de imperfecciones del sistema físico concreto empleado para representar estas cadenas binarias. El error que puede afectar a un bit consiste en que su valor sufra una permutación pasando de 0 a 1 o de 1 a 0 de manera indeseada. A este tipo de alteración del valor lo llamamos "inversión de bit" (*bit flip* en inglés) y es, evidentemente, el único tipo de error que puede darse en una cadena de bits[4]. Así pues, en computación clásica se consideran sinónimas las expresiones "error" e

[4]Asumimos que se conserva siempre el número de bits disponibles n. De no ser así, se podrían dar también los errores conocidos como "borrones", que consisten en la pérdida de un bit.

"inversión de bit".

El modelo clásico de errores consiste entonces en considerar que una cadena de bits puede verse alterada al sufrir uno de sus bits una inversión indeseada. Si la probabilidad de que esto ocurra toma un valor p entre 0 y 1, la teoría de códigos correctores clásicos estudia la manera de ampliar una cadena de bits con partes redundantes que permiten detectar y corregir un determinado número de errores, de tal modo que la probabilidad de que haya errores no detectados se reduzca a un nuevo valor p' tal que $p' < p$.

En la siguiente sección introduciremos los conceptos más básicos de la corrección clásica de errores. La relevancia de esta introducción radica en que la corrección cuántica de errores utiliza muchos conceptos análogos a los clásicos, por lo que resulta necesario tener al menos un mínimo conocimiento de la teoría clásica para poder comprender la investigación reciente en códigos correctores cuánticos.

7 Introducción a los códigos correctores clásicos

Como se ha comentado anteriormente, entendemos por "bit" un conjunto finito de dos elementos, $\{0, 1\}$ en la notación habitual. En este conjunto se puede definir una operación interna "suma módulo 2", con la tabla de operaciones $0 + 0 = 0$, $0 + 1 = 1$, $1 + 0 = 1$ y $1 + 1 = 0$. Con esta operación, a la que también se llama "o exclusivo lógico" (operación lógica *exclusive or* o *xor*), el conjunto de un solo bit tiene estructura de grupo, con la única ley de grupo posible en un conjunto finito de dos elementos. Se trata del grupo cíclico de dos elementos C_2.

Además de esta operación suma, podemos dotar al conjunto $0, 1$ de una segunda operación interna, a la que llamaremos "producto" o bien "y lógica" (operación lógica *and*) y cuya tabla de operaciones es $0 \cdot 0 = 0$, $0 \cdot 1 = 0$, $1 \cdot 0 = 0$ y $1 \cdot 1 = 1$. Con esta segunda operación interna, el conjunto finito de dos elementos adquiere estructura de cuerpo, el cuerpo finito (también llamado "cuerpo de Galois") de orden 2, para el que se utilizan diversas notaciones, como $GF(2)$, $\mathbb{Z}/2\mathbb{Z}$ o $\mathbb{F}_2$.

Así, con estas operaciones lógicas extremadamente simples hemos formalizado el concepto de bit como cuerpo finito de orden 2. Y es más, lo podemos considerar también dotado de una estructura trivial de espacio vectorial sobre el propio cuerpo $\mathbb{F}_2$.

Definición 7.1. Llamaremos **espacio de Hamming** de dimensión n al conjunto $\{0, 1\}^n = \mathbb{F}_2^n$. Este conjunto tiene estructura de espacio vectorial sobre el cuerpo $\mathbb{F}_2$. A los elementos del espacio de Hamming de dimensión n se les llama **cadenas de n bits** o **cadenas binarias de longitud** n o, también, **palabras de n bits**.

Veremos a continuación dos conceptos que servirán para especificar cuán diferentes son dos cadenas de n bits.

Definición 7.2. Dada una cadena de n bits x, se denomina **peso de Hamming** de x, $w(x)$, al número de componentes no nulas; es decir, al número de unos en la cadena.

Definición 7.3. Dadas dos cadenas de n bits, x e y, se denomina **distancia de Hamming** entre x e y al peso de la suma de las dos cadenas; es decir, $d(x, y) := w(x + y)$.

Nótese que esta distancia se define a partir del peso de la manera habitual en que se define una distancia a partir de una norma. Se utiliza $x + y$ en lugar del típico $x - y$ porque en un espacio de Hamming sumar y restar es lo mismo. Gracias a estas definiciones, el concepto intuitivo de cadena binaria se manifiesta como un elemento de un espacio vectorial dotado de una estructura métrica. Esta estructura métrica es esencial para poder cuantificar cuán cercanas o similares son dos cadenas binarias al analizar la aparición de errores sobrevenidos.

Para entender cómo surge la teoría formal de corrección clásica de errores, comentaremos en primer lugar de manera intuitiva el mecanismo de corrección más simple: el código de repetición.

Una de las formas más triviales de corregir posibles errores en una cadena binaria de longitud n consiste en repetir los valores de cada uno de los bits. La manera más sencilla de hacer esto consiste en repetir cada bit tres veces. Por ejemplo, una cadena de cuatro bits como 0-0-1-0 pasaría a representarse en forma de doce bits 000-000-111-000. ¿Qué ganamos con esto? Imaginemos que el mecanismo físico que representa la cadena de bits tiene unas características tales que la probabilidad de que cualquiera de los bits sufra una inversión después de una transmisión de información en el espacio o en el tiempo es un valor p $(0 < p < 1)$, igual e independiente para cada uno de los bits. Para reconstruir la cadena original habrá que recorrer los bits de la cadena modificada de tres en tres. Las posibilidades, para cada una de estas ternas, son ocho. En primer lugar, que ningún bit haya sufrido inversión; la probabilidad de este caso es $(1 - p)^3$. En segundo lugar, que uno de los tres bits haya sufrido inversión; cada uno de estos tres casos tiene probabilidad $p(1 - p)^2$. En tercer lugar, que dos de los tres bits hayan sufrido inversión; la probabilidad de estos otros tres casos sería $p^2(1 - p)$. Y, por último, el caso más desfavorable de que los tres bits se hayan invertido, cuya probabilidad sera p^3. Suponiendo que $p < 0,5$ (algo que se puede asumir sin pérdida de generalidad, pues en caso contrario, tendríamos un problema de calibración o definición de los estados 0 y 1 en el dispositivo físico y bastaría con reinterpretar los estados), entonces se cumple $(1 - p)^3 > p^3$ y $p(1 - p)^2 > p^2(1 - p)$. Esto implica que si nos encontramos los tres bits iguales 000 o 111, es más probable que no se haya producido ningún error a que se hayan producido tres inversiones. Y, del mismo modo, es más probable que los estados 100, 010, 001, 110, 101 y 011 correspondan al caso en que se ha invertido el bit diferente y no a que se hayan invertido los dos bits mayoritarios. Esto nos da una regla para corregir errores: supondremos que los estados 000, 100, 010 y 001 corresponden a un 0 original, mientras que los estados 111, 110, 101 y 011 corresponden a un 1 original. A este criterio de corrección del error se lo denomina "votación por

mayoría" (*majority voting*, en inglés) y reduce la probabilidad de error de inversión. En efecto, la probabilidad p de que se produzca un error de inversión en cada uno de los bits de la cadena ha pasado a ser, gracias a la repetición, $p^2(1-p)+p^3$. Luego para valores pequeños de p, el comportamiento pasa de ser de tipo $O(p)$ a $O(p^2)$. Si la probabilidad p^2 se juzgara aún excesiva, podría aplicarse el código de repetición de manera reiterada reduciéndola a p^4, p^8, etc. para 9, 27, etc. repeticiones.

Del ejemplo anterior se pueden extraer algunas características aplicables a otras formas de corrección de errores. Si la información original ocupa una cadena binaria de longitud n y no hay ningún tipo de redundancia en la información, entonces todos los bits son significativos y no habría manera de detectar una inversión de bit. La clave del procedimiento de repetición anterior reside en el paso a un espacio de dimensión superior que introduce redundancia en la información. Esto hace que se puedan hacer corresponder varias cadenas binarias del espacio mayor a cada una de las cadenas del espacio original. El proceso de corrección pasa entonces por tres fases: un paso de cadenas de longitud n a cadenas de longitud m con $m > n$, una transmisión espacial o evolución temporal del sistema en el espacio ampliado y, finalmente, una recuperación de la cadena en el espacio original. Estas ideas intuitivas dan lugar a definiciones estrictas de "código", "codificar" y "decodificar", que enunciamos a continuación.

Definición 7.4. Dado un espacio de Hamming de dimensión n, se llama **codificación** a una aplicación de este espacio de Hamming sobre otro de dimensión mayor m, al que se suele llamar **espacio de codificación**.

Definición 7.5. Dada una codificación de un espacio de Hamming de dimensión n sobre otro de dimensión m ($m > n$), se denomina **código** al conjunto imagen por la codificación.

A veces se aplica el adjetivo "binario" a estos códigos definidos sobre espacios de Hamming para diferenciarlos del concepto más general definido en espacios vectoriales sobre otros cuerpos finitos. Los códigos de Reed-Solomon son un ejemplo conocido de códigos no binarios. En el presente trabajo, contemplamos solamente los códigos binarios.

Definición 7.6. Dada una codificación entre espacios de Hamming, a los elementos del código se los denomina **palabras del código** (*codewords*, en inglés).

Definición 7.7. Dada una codificación entre espacios de Hamming, se llama **decodificación** a una aplicación suprayectiva del espacio de codificación de dimensión m sobre el espacio original de dimensión n ($m > n$).

Esta última aplicación de decodificación se descompone a veces en dos partes: una aplicación de recuperación mediante la que se recuperan las palabras del código y una segunda parte que coincide con la aplicación inversa de la codificación.

La identificación de estos conceptos en el caso de la triple repetición resulta obvia: si la cadena binaria original consta de un solo bit $\mathbb{F}_2 = \{0, 1\}$, tenemos un espacio de

Hamming original de dimensión 1. El espacio de codificación para la repetición sería el espacio de Hamming de dimensión 3 $\mathbb{F}_2^3 = \{000, 001, 010, 100, 011, 101, 110, 111\}$. La codificación es la aplicación dada por $0 \mapsto 000$ y $1 \mapsto 111$. El código de repetición es entonces el subconjunto del espacio de codificación cuyas palabras de código son $\{000, 111\}$. Y la decodificación es la aplicación correspondiente a la votación por mayoría ($000 \mapsto 0$, $001 \mapsto 0$, $010 \mapsto 0$, $100 \mapsto 0$, $011 \mapsto 1$, $101 \mapsto 1$, $110 \mapsto 1$, $111 \mapsto 1$).

Veamos ahora un teoremas importante que relaciona la distancia de Hamming con la capacidad correctora de un código [18, p. 42].

Teorema 7.1. *Sean un espacio de Hamming original de dimensión n, un espacio de codificación de dimensión m, $m > n$, y C un código en este espacio de codificación. Y sea $h = min_{u,v \in C}\{d(u, v)\}$; es decir, la mínima distancia de Hamming entre dos palabras del código. Entonces, este código puede detectar un máximo de $h - 1$ inversiones de bit. Además, el código C puede corregir un número de inversiones de bit menor que $h/2$.*

Demostración. Dadas dos palabras de código cuya distancia de Hamming es h, serán necesarias, por definición, h inversiones de bit para transformar una palabra en la otra, con lo que el error en una de las palabras sería indistinguible de la ausencia de error en la otra palabra. Por lo tanto, este código nunca podrá detectar en general h inversiones de bit. Pero si hay a lo sumo $h - 1$ inversiones, una palabra de código inicial no podrá convertirse en otra, de lo que se sigue que en tal circunstancia tienen que haberse producido inversiones de bit. Esto demuestra la primera parte del teorema.

Además, como h es la mínima distancia de Hamming que puede darse entre dos palabras de código, cualquier palabra que diste, en el sentido de Hamming, en menos de $h/2$ de una palabra de código, no podrá estar más cerca de ninguna otra (pues tendríamos una contradicción al sumar las distancias) por lo que habrá una manera natural de corregir las inversiones de bit sustituyendo cada una de estas palabras que haya sufrido inversiones por su palabra de código más cercana. □

En el caso del código de repetición triple, la mínima distancia de Hamming entre las palabras de código será la única que hay, entre 000 y 111, que es 3. Evidentemente, si tenemos una palabra en que aparecen tanto ceros como unos, necesariamente tiene que haberse producido un error y la distancia de Hamming entre las palabras alteradas y cada una de las palabras del código será de 1 y 2. Tomando la palabra de código a distancia 1; esto es, la que solamente difiere en un bit, obtenemos de nuevo el mecanismo de corrección que habíamos introducido anteriormente, ahora expresado más formalmente en función de la distancia de Hamming.

Introducimos ahora un tipo de notación habitual en la descripción de códigos clásicos que, como veremos más adelante, tiene un análogo en los códigos cuánticos.

Definición 7.8. Dado un código cuyo espacio de codificación tiene dimensión n y

que codifica un espacio original de dimensión k, se dice que es un código de tipo **(n, k)**. Si, además, la mínima distancia de Hamming entre las palabras del código es d, se dice que es de tipo **(n, k, d)**.

Con esta notación, el código de repetición triple aplicado a un solo bit es un código de tipo (3, 1, 3).

En general, las codificaciones son aplicaciones sobre espacios de Hamming que pueden responder a descripciones muy complejas. En el caso más extremo, dado que tratamos con conjuntos finitos, se puede definir un código enumerando las imágenes por la codificación de cada uno de los elementos del espacio de origen. Evidentemente, este planteamiento se vuelve inmanejable en cuanto crecen las dimensiones de los espacios. Tal como ocurre en general con las aplicaciones entre espacios vectoriales, la situación es muchísimo más simple cuando restringimos el estudio a aplicaciones lineales. En la siguiente definición, daremos nombre a este caso particular de códigos.

Definición 7.9. Se llama **código lineal** al código resultante de una codificación que es una aplicación lineal o endomorfismo del espacio de codificación.

El código de repetición triple es un código lineal, como se demuestra fácilmente evaluando la codificación para las sumas y productos posibles entre unos y ceros. La condición de linealidad permite aplicar a estos códigos todo el bagaje teórico del álgebra lineal. En particular, el código admite una representación matricial.

Definición 7.10. Dado un código lineal, se llama **matriz generatriz** del código a la matriz coordenada de la codificación como endomorfismo del espacio. A la matriz generatriz se la suele representar mediante la letra G.

En el caso del código de repetición triple, la matriz generatriz G sería una matriz columna con tres unos, como se comprueba fácilmente:

$$\begin{pmatrix}1\\1\\1\end{pmatrix}(0) = \begin{pmatrix}0\\0\\0\end{pmatrix} \qquad \begin{pmatrix}1\\1\\1\end{pmatrix}(1) = \begin{pmatrix}1\\1\\1\end{pmatrix} \tag{7.1}$$

La matriz generatriz nos permite codificar la cadena binaria del espacio original. Ahora bien, no nos proporciona ningún tipo de utilidad para detectar o corregir los posibles errores. Vamos a ver que hay una segunda matriz relacionada que sí permite *detectar* los errores sobrevenidos en el espacio de codificación. Introduciremos esta matriz primero como una idea feliz que funciona para la repetición triple (3, 1, 3). Esto nos ayudará a captar la idea intuitiva para poder después generalizar esta idea mediante una definición formal.

Sea la matriz binaria 2×3 siguiente:

$$H := \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \tag{7.2}$$

Veamos qué ocurre al aplicar esta matriz a las palabras del código de repetición triple:

$$\begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{7.3}$$

Y veamos qué ocurre cuando aplicamos esta matriz a los otros seis vectores del espacio de codificación:

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

$$\begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \tag{7.4}$$

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

La inspección de las igualdades matriciales (7.3) y (7.4) revela un resultado muy importante. La matriz (7.2) es un endomorfismo que encapsula un criterio de detección de errores. Cuando el vector en el espacio de codificación no ha sufrido inversiones de bit, la aplicación de esta matriz arroja el vector $(0, 0)$. Cuando es el primer bit el que habría sufrido inversión, según el criterio de voto por mayoría, entonces el resultado es $(1, 1)$. Y análogamente se obtienen $(1, 0)$ y $(0, 1)$ para inversiones en el segundo y tercer bit.

Así pues, vemos que mientras que la matriz G permite codificar el bit original, la matriz H permite detectar el tipo de error que se ha producido. Veamos ahora una definición formal para esta matriz H. Para ello, hemos de tener en cuenta dos ideas: en primer lugar, las igualdades (7.3) y (7.4) corresponden a una aplicación lineal cuyo núcleo coincide con el código. En segundo lugar, un resultado de álgebra lineal elemental garantiza que dado un espacio vectorial U y un subespacio vectorial $V \subset U$ siempre se puede definir un endomorfismo f en U tal que $\ker f = U$. Esto implica que una matriz como H tiene que existir para cualquier otro código lineal y podemos utilizar como definición general esta idea del endomorfismo cuyo núcleo es el código. Combinando estas ideas, obtenemos la definición siguiente:

Definición 7.11. Dado un código lineal C, se llama **endomorfismo de control** del código a un endomorfismo H definido en el espacio de codificación tal que $\ker H = C$. Análogamente, a la matriz coordenada del endomorfismo se la denomina **matriz de control**.

Cuando se aplica H a una cadena binaria del espacio de codificación que no es parte del código, entonces el resultado será diferente de cero y puede dar información sobre las inversiones de bit concretas que se han producido. Dada la importancia de estos valores, conviene darles un nombre.

Definición 7.12. Dado un endomorfismo de control H de un código lineal, a las imágenes por H de los vectores del espacio de codificación se las denomina **síndromes**.

En el caso del código de repetición triple, los síndromes serían $(0, 0)$ (ausencia de error), $(1, 1)$ (inversión del primer bit), $(1, 0)$ (inversión del segundo bit) y $(0, 1)$ (inversión del tercer bit).

Así pues, dado un código lineal podremos formar estas dos matrices G y H. Como G tiene como imagen el subespacio código, que es precisamente el núcleo de H, se cumplirá la siguiente proposición:

Proposición 7.2. *Dado un código lineal C, sus matrices generatriz y de control cumplen la siguiente relación:*

$$HG = 0$$

Demostración. Dado que G representa un endomorfismo cuya imagen es el código y H tiene precisamente al código como núcleo, la aplicación compuesta será idénticamente nula. Luego su matriz coordenada es la matriz nula. □

En la literatura sobre códigos clásicos se pueden encontrar muchas más propiedades de estas matrices y maneras constructivas de determinar una a partir de la otra que exceden el propósito de este trabajo[5]. Apuntaremos únicamente como último resultado que si reemplazamos el código por su subespacio complementario en el espacio de codificación, entonces los papeles de las matrices generatriz y de control se invierten. Esto conduce a una idea de dualidad. Dado un código lineal cuyas matrices generatriz y de control son G y H respectivamente, a su subespacio complementario se le llama "código lineal dual". Se demuestra fácilmente que la matriz generatriz del dual es H^T y la de control G^T.

Acabamos esta pequeña introducción a la teoría clásica de códigos correctores mencionando otro código lineal muy conocido y documentado: el código de Hamming (7, 4). La descripción detallada de los códigos de Hamming excedería también los límites de este trabajo; basta con saber que los códigos de Hamming son una familia de códigos lineales muy útiles en los que las cadenas binarias de longitud n se

[5]Una obra con información muy detallada sobre estos aspectos es *Codificación de la información*, de Carlos Munuera y Juan Gabriel Tena [18].

reemplazan por cadenas en que los datos originales se extienden con un número adicional de bits que representan las paridades de las subcadenas de longitud $n-1$. La "paridad" de una cadena binaria se define como la condición de que el número de unos en la cadena sea par o impar (más formalmente se podría definir como el peso de la cadena módulo 2). Estos bits de paridad ocupan las posiciones correspondientes a potencias de dos en la cadena (1, 2, 4, 8, 16, etc.) hasta donde sea posible. El procedimiento de detección de errores consiste en evaluar los bits de paridad de la cadena susceptible de haber sufrido errores y comprobar si se ajustan a los bits de datos. En caso de que haya discrepancias, necesariamente tienen que haberse producido errores.

La manera de distribuir los bits de paridad hace que el código de repetición triple sea un caso particular de código de Hamming, el correspondiente al caso (3, 1, 3). El caso más simple de código de Hamming que no es de repetición es el código de tipo (7, 4) (se podría utilizar también la notación (7, 4, 3) pero la mínima distancia de Hamming entre las palabras de estos códigos es siempre 3, por lo que es innecesario), en el que se codifica una cadena binaria de longitud 4 añadiendo tres bits de paridad para las tres subcadenas de longitud 3 que surgen de fijar uno de los cuatro bits y combinar los otros dos entre los tres bits restantes (las demás paridades serían deducibles de estas).

El código de Hamming (7, 4) tiene las siguientes matrices generatriz y de control:

$$G := \begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad H := \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix} \tag{7.5}$$

Tras esta breve introducción a los códigos correctores clásicos, en las secciones siguientes nos centraremos ya únicamente en la corrección cuántica y veremos que estos códigos lineales clásicos que hemos introducido aquí tienen sus análogos en el caso cuántico.

8 Modelo de errores en computación cuántica

El modelo teórico consistente en un sistema cerrado de uno o varios *qubits* que evolucionan por la aplicación de transformaciones unitarias es evidentemente una simplificación, ya que en la naturaleza no hay ningún sistema realmente cerrado aparte del universo en su totalidad. Esto implica que en cualquier dispositivo físico en el que se configure un número n de *qubits* sobre los que actuarán determinadas transformaciones unitarias y procesos de medición, existirá siempre la posibilidad

de que se manifiesten efectos imprevistos en los que participa el entorno.

Por ejemplo, podrían darse transformaciones unitarias indeseadas e incluso el sistema de n *qubits* podría no ser tal al interactuar con *qubits* adicionales. Además, no podemos garantizar con total certidumbre que los estados iniciales de los *qubits* serán los deseados. En el caso general, tendremos que utilizar los estados mixtos descritos por operadores de densidad que hemos introducido anteriormente.

Al conjunto de estas interacciones indeseadas con el entorno se las denomina de manera genérica "ruido" y se trata de un problema tanto en la computación clásica como en la computación cuántica, aunque en esta última se manifiesta de una manera más clara ya que los estados de un *qubit* forman un continuo, a diferencia de lo que ocurre con los bits clásicos. Esto, unido a la propia dificultad de manipular sistemas físicos de tamaño ínfimo para que se manifiesten los fenómenos cuánticos, hace del ruido cuántico un problema más difícil de tratar que su análogo clásico.

Una primera aproximación a la descripción de errores consiste en asumir que los errores se manifiestan como transformaciones unitarias indeseadas sobre *qubit*s aislados. Este caso simplificado presenta además la ventaja de ser el que muestra mayor analogía con el tratamiento convencional de los errores clásicos.

8.1 Errores en un sistema de un *qubit*

Si asumimos que nuestro universo consta de un único *qubit*, su estado en un instante de tiempo inicial t_0 vendrá dado por un vector $|\psi_0\rangle$ del espacio de Hilbert H de estados. En la base $\{|0\rangle, |1\rangle\}$ podemos expresar este vector como $|\psi_0\rangle = a_0|0\rangle + b_0|1\rangle$, con $a_0, b_0 \in \mathbb{C}$ tales que $|a_0|^2 + |b_0|^2 = 1$.

La teoría cuántica nos dice que en un instante de tiempo posterior t_1, el estado del sistema será un vector $|\psi_1\rangle = a_1|0\rangle + b_1|1\rangle$ de tal modo que este estado sea el resultado de aplicar un operador unitario al estado original $|\psi_0\rangle$; es decir que existe un operador unitario U tal que $|\psi_1\rangle = U|\psi_0\rangle$. Esto incluye como caso trivial la posibilidad de que el estado no haya variado, cuando el operador unitario es la identidad.

La situación general es que en una situación en que esperaríamos no tener ningun cambio de estado:

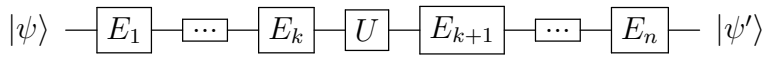
$$|\psi\rangle \longrightarrow |\psi\rangle \tag{8.1}$$

se cuela en realidad una transformación indeseada E:

$$|\psi\rangle \longrightarrow \boxed{E} \longrightarrow |\psi'\rangle \tag{8.2}$$

En el caso de un sistema de un único *qubit*, las transformaciones unitarias se pueden representar como matrices unitarias 2×2 que, como vimos en la sección 2, se pueden expresar como combinación lineal de las matrices de Pauli σ_x, σ_y, σ_z y la identidad I. Así pues, la transformación de error E será una combinación lineal de estas cuatro matrices.

Veamos entonces qué ocurre en este caso simplificado cuando se hace pasar un *qubit* por una puerta lógica cuántica. Dado que la puerta lógica es matemáticamente la aplicación de una transformación unitaria, los errores sobre este sistema consistirán en transformaciones unitarias indeseadas adicionales, como se muestra en la figura siguiente:

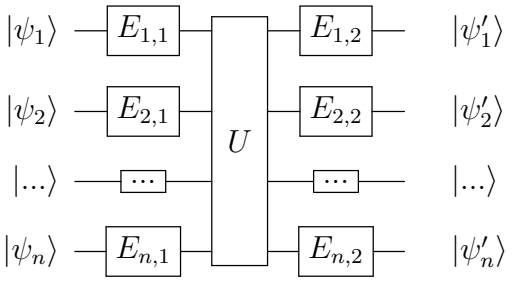


(8.3)

En el diagrama anterior asumimos que aparece un total de n transformaciones unitarias indeseadas E_i, de las que un número k transforman el *qubit* antes de que entre por la transformación deseada U, pero como el producto de transformaciones unitarias es también una transformación unitaria, sin pérdida de generalidad se pueden agrupar las transformaciones contiguas y considerar que un sistema cuántico evoluciona en pasos discretos en los que junto a cada transformación unitaria que es parte del diseño de la puerta lógica puede aparecer una transformación indeseada. El problema que se nos plantea consiste entonces en ser capaces de detectar situaciones como la de (8.2) y recuperar el estado sin error de (8.1).

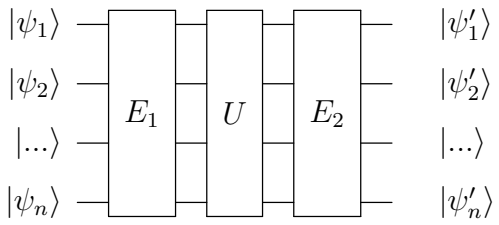
8.2 Errores en un sistema de múltiples *qubits*

Si tenemos un sistema formado por un número finito $n \in \mathbb{N}$ de *qubits*, podemos considerar, en una primera aproximación, que los errores afectan a cada *qubit* de manera individual, tal como se ha descrito en el apartado anterior.



(8.4)

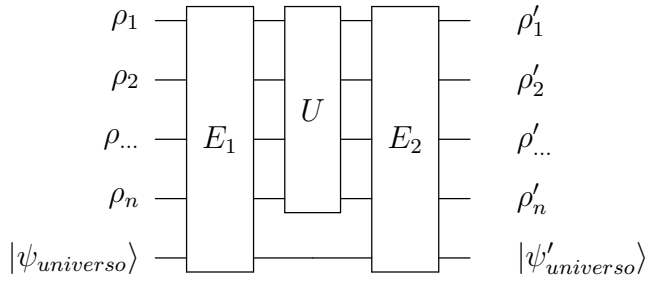
Un tratamiento más complejo manteniendo la simplificación de que el sistema de n *qubits* está totalmente aislado del entorno, consiste en tener en cuenta las interacciones entre *qubits*. Sería el modelo de errores que se representa en la siguiente figura:



(8.5)

8.3 Errores en el caso más general

En el caso más general, los n *qubits* que forman el dispositivo de computación cuántica interactuarán con el resto del Universo, con lo que sus estados tendrán que describirse necesariamente mediante operadores de densidad que reflejen la incertidumbre asociada a tales interacciones con el entorno, tal como se explicó en la introducción al formalismo matemático de la sección 2. En el caso en el que tenemos n *qubits* a los que se aplica una transformación unitaria U, la situación general sería la de la siguiente figura:



(8.6)

9 Introducción a la corrección cuántica de errores

Como se ha comentado en la sección anterior, los sistemas físicos que subyacen a un ordenador cuántico están sometidos inevitablemente a interferencias con el exterior en forma de decoherencia y ruido. Notemos que mientras que la decoherencia es un problema que aparece solamente en el caso cuántico, el problema del ruido afecta también a los ordenadores clásicos en forma de inversiones de bit indeseadas. Por ello, al plantear el diseño de códigos correctores cuánticos la primera idea que surge es la de intentar aplicar ideas de la corrección clásica de errores.

¿Se podría aplicar la técnica del código de repetición clásico que explicamos en una sección anterior al caso cuántico? Es decir, ¿podríamos tomar un sistema de n *qubits* $|101\ldots0\rangle$ y convertirlo en $|111000111\ldots000\rangle$? La respuesta es que sí se podría en este caso en el que el estado de los *qubits* está compuesto por estados observables $|0\rangle$ y $|1\rangle$, pero esto sería el equivalente al caso clásico y nos interesa ir más allá; querríamos poder replicar *qubits* en estados arbitrarios de superposición $|\psi_1\psi_2\psi_3...\psi_n\rangle$, para preparar el estado $|\psi_1\psi_1\psi_1\psi_2\psi_2\psi_2\psi_3\psi_3\psi_3...\psi_n\psi_n\psi_n\rangle$. De manera

un tanto sorprendente, hoy sabemos que esto es imposible. Este resultado se conoce formalmente como el "teorema de la imposibilidad de clonación" (*no cloning theorem*, en inglés):

Teorema 9.1. *(Teorema de la imposibilidad de clonación) Dado un estado cuántico $|\psi\rangle$, es imposible clonar el estado para obtener $|\psi\psi\rangle$ [13] [2].*

Demostración. Dado que los sistemas de *qubits* se transforman mediante endomorfismos unitarios, para conseguir un estado $|\psi\psi\rangle$ en un espacio de estados $H^{\otimes 2}$ a partir de un estado $|\psi\rangle$ en un espacio de estados H de n *qubits*, será necesario preparar el espacio H^2 con un estado inicial $|\psi\rangle \otimes |0\rangle^{\otimes n}$ (asumimos siempre que es posible preparar estados $|0\rangle$) y definir un endomorfismo unitario C en H^2 tal que $C(|\psi\rangle \otimes |0\rangle^{\otimes n}) = |\psi\rangle \otimes |\psi\rangle$ para todo estado $|\psi\rangle$ de H. Si suponemos que existe un endomorfismo unitario C que cumpla esta propiedad de clonación, se tendrá que:

$$\begin{aligned} C(|0\rangle \otimes |0\rangle) &= |0\rangle \otimes |0\rangle \\ C(|1\rangle \otimes |0\rangle) &= |1\rangle \otimes |1\rangle \end{aligned} \tag{9.1}$$

Si ahora aplicamos C al estado $|+\rangle := \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, tendremos, por definición de C:

$$C(|+\rangle \otimes |0\rangle) = |+\rangle \otimes |+\rangle = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle) \tag{9.2}$$

Dado que C es lineal, deberíamos poder también calcular $C(|+\rangle \otimes |0\rangle)$ a partir de las expresiones de (9.1). De esa manera, obtenemos:

$$\begin{aligned} C(|+\rangle \otimes |0\rangle) = C(\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |0\rangle) = \\ = \frac{1}{\sqrt{2}}(C(|0\rangle \otimes |0\rangle) + C(|1\rangle \otimes |0\rangle)) = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \end{aligned} \tag{9.3}$$

Los resultados obtenidos en (9.2) y (9.3) son diferentes, por lo que hemos llegado a una contradicción. Luego no puede existir el endomorfismo C postulado. □

Aunque puede parecer un resultado sorprendente, se puede captar el sentido intuitivo del *no cloning theorem* si pensamos que el mero hecho de copiar un *qubit* implicaría su observación para poder recrear el resultado de la medición. Pero hemos visto que uno de los postulados de la mecánica cuántica afirma que la medición distorsiona inevitablemente el estado original, haciéndolo decaer a un autoestado del operador lineal asociado. Así pues, la copia de un sistema de *qubits* sería equivalente a su observación y la estrategia ingenua de repetir *qubits* resulta ser teóricamente imposible.

Tras este primer intento fallido de extender al caso cuántico un tipo sencillo de código corrector clásico, veamos las dificultades principales a las que nos enfrentamos para corregir los errores en computación cuántica. Siguiendo la exposición de Nielsen y Chuang [21], hay tres tipos de dificultades principales:

1. **El teorema de la imposibilidad de clonación**. Como acabamos de ver, no es posible replicar *qubits* arbitrarios.

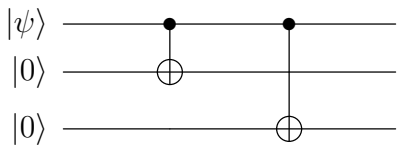
2. **La continuidad de los errores cuánticos**. Mientras que un bit clásico tiene dos valores posibles 0 y 1, el estado general de un *qubit* puede sufrir un continuo de distorsiones, como por ejemplo un pequeño incremento en una fase: un estado preparado como $\frac{1}{\sqrt{2}}(|0\rangle + e^{\phi}|1\rangle)$ podría acabar como $\frac{1}{\sqrt{2}}(|0\rangle + e^{\phi+\epsilon}|1\rangle)$.

3. **La decoherencia**. Como comentamos en la sección anterior, la observación de un estado no tiene por qué ser una medición llevada a cabo conscientemente, sino que cualquier pequeña interacción accidental indeseada con el sistema puede equivaler a una observación que haga colapsar el estado de los *qubits*.

9.1 El código corrector de inversiones de bit

Como acabamos de ver en la sección anterior, al no ser posible replicar el estado de un *qubit*, no podemos pasar de un estado de superposición arbitrario $|\psi\rangle$ a $|\psi\psi\psi\rangle$. Pero no todo está perdido, vamos a ver que en realidad sí es posible un tipo más sutil de repetición, propuesto por primera vez por Peter Shor.

Recordemos que la notación $|\psi\psi\rangle$ es simplemente una abreviatura de un producto tensorial $|\psi\rangle \otimes |\psi\rangle$ y que un estado de superposición cualquiera $|\psi\rangle$ de un *qubit* se podrá expresar como $a|0\rangle + b|1\rangle$, en donde a y b son números complejos tales que $|a|^2 + |b|^2 = 1$. Un tipo de repetición que sí es posible consiste en construir el estado de tres *qubits* $a|000\rangle + b|111\rangle$. La manera de construir este estado consiste en una puerta lógica formada por dos puertas CNOT en las que se utiliza el *qubit* $|0\rangle$ como segunda entrada [21, p. 428]:



(9.4)

Nótese que para pasar de un sistema de un *qubit* a otro de tres *qubits* mediante una transformación unitaria necesitamos complementar el *qubit* original con dos *qubits* auxiliares (o "ancilares", *ancilla qubits* en inglés). Asumiendo que es tecnológicamente posible suplementar los N *qubits* originales con tantos *qubits* ancilares como se necesiten, podemos dar por factibles este tipo de transformaciones.

Mediante la puerta lógica que acabamos de construir, se podrá entonces convertir un sistema de un *qubit* (espacio de dimensión 2) en un sistema de tres *qubits* (espacio

de dimensión 8) reemplazando respectivamente los vectores base $|0\rangle$ y $|1\rangle$ por los *qubits* $|000\rangle$ y $|111\rangle$.

Este tipo de repetición permite corregir el equivalente cuántico de la inversión de bit, que se da cuando en el vector de estado de un *qubit* se invierte el papel de los dos *qubits* base $|0\rangle$ y $|1\rangle$. Podemos definir formalmente este tipo de error como una transformación unitaria sobre el espacio de Hilbert de un solo *qubit*.

Definición 9.1. Dado un sistema de un *qubit*, se llama **inversión de bits** (*bit flip*, en inglés) al endomorfismo unitario X que a un estado $|\psi\rangle = a|0\rangle + b|1\rangle$ le hace corresponder el estado $X|\psi\rangle = b|0\rangle + a|1\rangle$.

La matriz coordenada de la inversión de bits en la base $\{|0\rangle, |1\rangle\}$ será:

$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \tag{9.5}$$

Esta transformación es exactamente la misma que la matriz de Pauli σ_x que vimos en una sección anterior. Podemos plantear el problema en los siguientes términos: dado un sistema de un *qubit*, su estado $|\psi\rangle$ sufre en el transcurso de un intervalo de tiempo t una transformación σ_x con probabilidad p ($0 \leqslant p \leqslant 1$). Transcurrido ese tiempo, ¿podemos diagnosticar si se ha producido esa transformación indeseada? Veamos lo que ocurre en el espacio de codificación de dimensión 8. Primero, transformamos el estado al espacio de codificación:

$$|\psi\rangle = a|0\rangle + b|1\rangle \mapsto a|000\rangle + b|111\rangle \tag{9.6}$$

Ahora se deja transcurrir el tiempo t en que el estado es susceptible de sufrir la transformación X en cada uno de sus *qubits* con probabilidad p (asumimos que los *qubits* son exactamente iguales en sensibilidad a errores). Se pueden producir entonces ocho evoluciones posibles:

1. Los estados $|000\rangle$ y $|111\rangle$ se mantienen invariables con probabilidad $(1-p)^3$.

2. Los estados $|000\rangle$ y $|111\rangle$ se transforman en $|100\rangle$ y $|011\rangle$ respectivamente con probabilidad $p(1-p)^2$.

3. Los estados $|000\rangle$ y $|111\rangle$ se transforman en $|010\rangle$ y $|101\rangle$ respectivamente con probabilidad $p(1-p)^2$.

4. Los estados $|000\rangle$ y $|111\rangle$ se transforman en $|001\rangle$ y $|110\rangle$ respectivamente con probabilidad $p(1-p)^2$.

5. Los estados $|000\rangle$ y $|111\rangle$ se transforman en $|110\rangle$ y $|001\rangle$ respectivamente con probabilidad $p^2(1-p)$.

6. Los estados $|000\rangle$ y $|111\rangle$ se transforman en $|101\rangle$ y $|010\rangle$ respectivamente con probabilidad $p^2(1-p)$.

7. Los estados $|000\rangle$ y $|111\rangle$ se transforman en $|011\rangle$ y $|100\rangle$ respectivamente con probabilidad $p^2(1-p)$.

8. Los estados $|000\rangle$ y $|111\rangle$ se transforman en $|111\rangle$ y $|000\rangle$ respectivamente con probabilidad p^3.

La situación es muy similar a la que hemos visto con el código clásico de repetición. La probabilidad p' de que la inversión de bit se produzca en dos o tres de los *qubits* será la suma de las probabilidades de los cuatro últimos casos: $p' = 3p^2(1-p)+p^3 = 3p^2 - 2p^3$, que es menor que p cuando $p < 0,5$. Además, mientras que en el espacio de Hilbert original el comportamiento asintótico de esta probabilidad es $O(p)$, en el espacio de codificación es $O(p^2)$. Así pues, la probabilidad $1-p$ de que el sistema original de un *qubit* no sufra error se convertirá en el espacio de codificación en una probabilidad mayor $1-p'$ de que el sistema se mantenga en uno de los cuatro primeros casos en los que hay a lo sumo un *qubit* que sufre inversión.

En la introducción a la corrección clásica de errores vimos cómo estos casos posibles podían detectarse comprobando los llamados síndromes, que eran las imágenes de los vectores por el endomorfismo de control. En este caso cuántico, existe un procedimiento análogo basado en endomorfismos hermíticos, que como hemos visto corresponden a magnitudes observables en la teoría cuántica. Los síndromes vendrán dados ahora por cuatro endomorfismos hermíticos que permitan detectar si las coordenadas no nulas del vector son las que corresponden bien a $|000\rangle$ y $|111\rangle$ o bien a $|100\rangle$ y $|011\rangle$ o bien a $|010\rangle$ y $|101\rangle$ o bien a $|001\rangle$ y $|110\rangle$. Los endomorfismos necesarios serán los siguientes:

1. $P_0 := |000\rangle\langle 000| + |111\rangle\langle 111|$. Si $\langle\phi|P_0|\phi\rangle = 1$, entonces se trata del caso 1 en que no ha habido alteración.

2. $P_1 := |100\rangle\langle 100| + |011\rangle\langle 011|$. Si $\langle\phi|P_1|\phi\rangle = 1$, entonces se trata del caso 2 en que se ha invertido el primer *qubit*.

3. $P_2 := |010\rangle\langle 010| + |101\rangle\langle 101|$. Si $\langle\phi|P_2|\phi\rangle = 1$, entonces se trata del caso 3 en que se ha invertido el segundo *qubit*.

4. $P_3 := |001\rangle\langle 001| + |110\rangle\langle 110|$. Si $\langle\phi|P_3|\phi\rangle = 1$, entonces se trata del caso 4 en que se ha invertido el tercer *qubit*.

Evidentemente, estas aplicaciones nos darían un diagnóstico erróneo cuando haya habido más de una inversión de bit, pero ese es el caso que hacemos probabilísticamente mucho más infrecuente gracias al código de repetición.

Una vez diagnosticado el error mediante estos cálculos de síndrome, la corrección del error en los tres casos en que ha habido inversión requerirá aplicar también un operador hermítico. Hemos visto que la inversión de bit en un sistema de un *qubit*

es el endomorfismo unitario correspondiente a la matriz de Pauli σ_x, por lo que las inversiones de cada *qubit* en el sistema de tres *qubits* corresponderán a productos tensoriales de σ_x con la identidad; en concreto:

1. En el caso 1 en que no hay alteración, la recuperación se efectúa obviamente sin hacer absolutamente nada. El endomorfismo de corrección es simplemente la identidad I.

2. En el caso 2 en que se invierte el primer *qubit* la recuperación requiere invertir de nuevo el *qubit* afectado. El endomorfismo de corrección es $\sigma_x \otimes I \otimes I$.

3. En el caso 3 en que se invierte el segundo *qubit* la recuperación requiere invertir de nuevo el *qubit* afectado. El endomorfismo de corrección es $I \otimes \sigma_x \otimes I$.

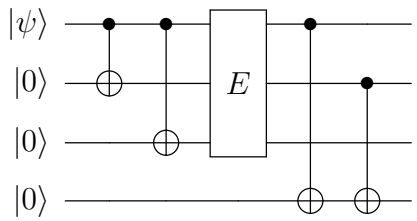
4. En el caso 4 en que se invierte el tercer *qubit* la recuperación requiere invertir de nuevo el *qubit* afectado. El endomorfismo de corrección es $I \otimes I \otimes \sigma_x$.

De esta manera, el estado del sistema en el espacio de codificación vuelve a estar contenido en el subespacio generado por $\{|000\rangle, |111\rangle\}$ y podemos pasarlo al espacio original de un solo *qubit* simplemente ignorando los *qubits* ancilares.

Como en el caso del código de repetición clásico, este análogo cuántico puede aplicarse de manera reiterada pasando sucesivamente a sistemas de 9, 27, 81, etc. *qubits*. En general, en cada reiteración i-ésima de la codificación pasaríamos a un espacio de 3^i *qubits* en el que la probabilidad de errores de inversión de bit se reduce de $p_i \leqslant 0,5$ a $p_{i+1} = 3p_i^2 - 2p_i^3$.

Un problema con el procedimiento anterior es la dependencia de la medición de cuatro observables, lo que hace que no se pueda trasladar de manera inmediata al lenguaje de puertas lógicas. Una manera alternativa de identificar la posibilidad de una inversión de bit en los tres *qubits* consiste en añadir más *qubits* ancilares con los que poder identificar dónde puede haberse producido la inversión. Veamos paso a paso cómo hacer esto.

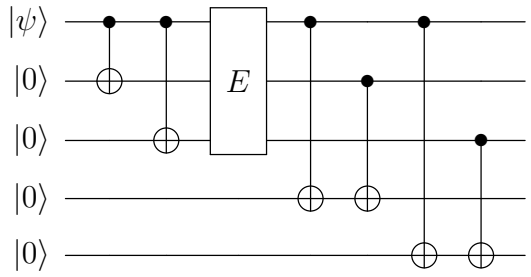
En primer lugar, si añadimos un *qubit* ancilar adicional y dos puertas CNOT podemos detectar si los dos primeros *qubits* son iguales (estado $|00\rangle$ o $|11\rangle$) o diferentes (estado $|10\rangle$ o $|01\rangle$):



(9.7)

Estas dos puertas CNOT hacen que el cuarto *qubit* inicializado como $|0\rangle$ se mantenga como $|0\rangle$ cuando el estado de los dos *qubits* iniciales sea $|00\rangle$ o $|11\rangle$, mientras que pasará al estado $|1\rangle$ cuando el estado de los dos *qubits* iniciales sea $|10\rangle$ o $|01\rangle$.

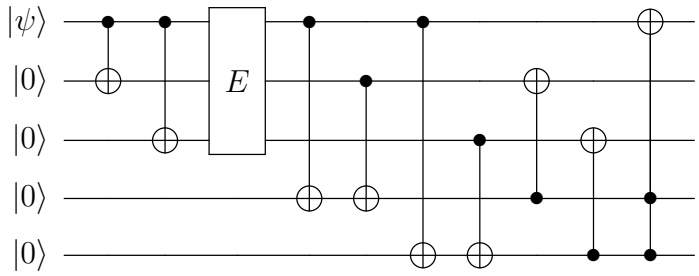
De esta manera, el nuevo *qubit* ancilar determina si puede haberse producido una inversión de bit en los dos primeros *qubits*. Para determinar si se ha producido una inversión de bit en el conjunto de los tres *qubits* bastará con añadir otro *qubit* ancilar análogo que dependa del tercer *qubit*:



(9.8)

Gracias a este quinto *qubit*, el circuito determina ahora también si coinciden o difieren los estados del primer y el tercer *qubit*. Esto nos permite ya llevar a cabo la detección completa del síndrome: si no se ha producido alteración, los tres *qubits* serán $|000\rangle$ o $|111\rangle$ y los estados de los dos últimos *qubits* ancilares serán ambos $|0\rangle$. Si se ha producido inversión del primer *qubit*, entonces los estados de estos dos últimos *qubits* serán ambos $|1\rangle$. En caso de inversión en el segundo *qubit* tendremos $|1\rangle$ y $|0\rangle$; y, finalmente, si ha sufrido inversión el tercer *qubit*, $|0\rangle$ y $|1\rangle$.

Para completar el circuito corrector, solamente nos resta añadir puertas de inversión de bit σ_x (o simplemente X, en la notación habitual en circuitos) que estén controladas por los estados de estos dos últimos *qubits* ancilares. Observemos que una puerta X controlada es exactamente lo mismo que la puerta CNOT. En el caso de que sea el primer *qubit* el que ha sufrido la inversión de bit, se necesitará un doble control, invirtiéndolo cuando los dos *qubits* ancilares estén en el estado $|1\rangle$; este control mediante doble condición corresponde a la puerta de Toffoli que vimos en la sección 3. El circuito final será entonces el que se muestra en la figura siguiente:



(9.9)

En el circuito anterior, se asume que los *qubits* ancilares cuarto y quinto se preparan en el estado $|0\rangle$ después de que haya transcurrido el tiempo durante el cual puede sobrevenir el error, que solamente afecta a los tres *qubits* del espacio de codificación.

El circuito (9.9) es el ejemplo más sencillo de circuito corrector, pero solamente corrige errores de inversión de bit. En una sección posterior, veremos otros códigos más sofisticados y para otros tipos de errores. Pero, antes, aprovecharemos las

conclusiones de este primer caso que hemos analizado para formalizar algunas definiciones de los conceptos principales y deducir algunos resultados que nos servirán para construir códigos más potentes.

9.2 Formalización de la teoría cuántica de corrección de errores

Tras haber introducido de manera intuitiva la construcción de algunos códigos correctores sencillos, vamos a ver cómo podemos formalizar los conceptos vistos. Empezaremos definiendo los espacios en los que se trabaja, siguiendo un planteamiento análogo al que vimos para la corrección clásica. La diferencia esencial radica en que en el caso cuántico se trabaja con espacios de Hilbert en lugar de los espacios de Hamming con los que trabajábamos en la teoría clásica.

Definición 9.2. Se llama **espacio original** al espacio de Hilbert H_0 en el que se encuentra el conjunto de *qubits* que participan en las operaciones lógicas.

En el ejemplo que hemos visto del código de repetición, el espacio original H_0 es el sistema de un único *qubit*.

Definición 9.3. Dado un espacio original H_0, se llama **espacio de codificación** (o **espacio de errores**) a un espacio de Hilbert H_e, obtenido a partir de H_0, en el que se produce la evolución que puede estar sujeta a errores.

En el caso del código de repetición, el espacio de codificación será el espacio de Hilbert de dimensión 8. Hasta aquí hemos estado utilizando la expresión "código corrector" como una manera genérica de denominar a un procedimiento para corregir errores. Vamos a pasar a definir el concepto de manera rigurosa, análogamente a como se definió en la teoría clásica.

Definición 9.4. Dado un espacio original H_0 y un espacio de codificación H_e se llama **código corrector** a un subespacio de Hilbert H_c de H_e de la misma dimensión que el espacio original H_0 y que habrá de cumplir ciertas condiciones que veremos más adelante.

Definición 9.5. A los elementos de una base ortonormal del código corrector se les denomina habitualmente **palabras del código**.

En el caso del código de repetición, el código corrector en este sentido formal es el subespacio de dimensión 2 generado por las palabras $\{|000\rangle, |111\rangle\}$.

El proceso de corrección de errores consistirá entonces en establecer aplicaciones entre estos espacios, para las que daremos nombres a continuación.

Definición 9.6. Llamamos **codificación** a un homomorfismo entre espacios vectoriales $C : H_0 \to H_c$ que hace corresponder a cada vector de H_0 un vector del código

corrector H_c. Esta aplicación lineal ha de ser unitaria para que el paso a H_c no afecte a las mediciones con significado físico.

Así pues, el sistema descrito originalmente en H_0 se encontrará entonces en un estado inicial determinado por un vector de estado puro $|\psi_0\rangle$. Al transformar este estado mediante la codificación, pasa a estar descrito en el espacio de Hilbert H_c por el estado $|\psi_c\rangle := C|\psi_0\rangle$.

De manera más general (para incluir también estados mixtos), podemos tratar los estados del sistema utilizando operadores de densidad en el espacio $E(H_0)$ de endomorfismos sobre H_0. La aplicación C puede extenderse de manera trivial al espacio de endomorfismos definiendo una aplicación $\tilde{C} : E(H_0) \to E(H_c)$ tal que la imagen de un estado $\rho_0 \in E(H_0)$ sea $\tilde{C}(\rho_0) = C\rho_0 C^\dagger$. Al estado $\tilde{C}(\rho_0)$ lo representaremos de manera más sencilla como ρ_c.

El estado $|\psi_c\rangle$ está definido en H_c. Por medio de la aplicación canónica de inclusión en el superespacio podemos transformarlo en el estado $|\psi_e\rangle$ definido en el espacio de codificación H_e. Análogamente, el operador de densidad ρ_c del espacio de endomorfismos $E(H_c)$ se transforma, por la aplicación canónica de inclusión, en el operador ρ_e como endomorfismo en $E(H_e)$.

Lo importante en este periplo por espacios de Hilbert es el hecho de que en ningún momento hemos alterado las características físicas observables del sistema. Al tratarse de aplicaciones unitarias, cualquier medición de observables arrojará los mismos resultados en el sistema original H_0, en el código H_c y en el espacio de codificación H_e. La idea que estamos desarrollando consiste pues en trasladar la descripción del sistema de un espacio de Hilbert original a otro físicamente equivalente, pero de dimensión mayor, en el que la probabilidad de errores indeseados se reduzca. Como hemos visto en el ejemplo del código de repetición, esto es de hecho posible. En aquel caso, dedujimos que una probabilidad de error $p < 0,5$ del espacio H_0 cuya base es $\{|0\rangle, |1\rangle\}$ se reduciría a $3p^2 - 2p^3$ en el espacio H_e, en ese caso el generado por $\{|000\rangle, |001\rangle, |010\rangle, |011\rangle, |100\rangle, |101\rangle, |110\rangle, |111\rangle\}$. Esta idea es la que queremos generalizar a sistemas más complicados y a otros tipos de errores.

En el estudio de errores, podemos considerar sin pérdida de generalidad que la evolución temporal, que definimos formalmente en la sección 2, consta únicamente de dos elementos: el estado inicial $|\psi_0\rangle$, "antes" de que se produzca el posible error, y el estado final $|\psi_1\rangle$, "después" de que se haya producido el posible error. En adelante, asumiremos esta evolución temporal simplificada a dos estados y nos referiremos simplemente a los estados inicial y final del sistema o al estado del sistema antes y después de la evolución.

Así pues, si el estado inicial del sistema es un estado puro $|\psi_0\rangle_0$ en el espacio H_0 y un estado puro $|\psi_0\rangle_e$ en el espacio H_e, el estado final será $|\psi_1\rangle_0 = U_0|\psi_0\rangle_0$ en el espacio original H_0 y $|\psi_1\rangle_e = U_e|\psi_0\rangle_e$ en el espacio de codificación H_e. Es este último estado en el espacio de codificación el que nos ha de permitir corregir ciertos errores conocidos. En el ejemplo que estamos tomando como referencia, la

inversión de bit, el estado $|\psi_0\rangle_e$ se encontraba en el plano generado por $\{|000\rangle, |111\rangle\}$ y ahí permanecía en ausencia de error, pero la posibilidad de una inversión de bit en cada uno de los tres *qubits* hacía que el estado pasara a otros subespacios de H_e; en particular, al plano generado por $\{|100\rangle, |011\rangle\}$ cuando hay inversión del primer *qubit* y a los planos generados por $\{|010\rangle, |101\rangle\}$ y $\{|001\rangle, |110\rangle\}$ cuando hay inversión del segundo y tercer *qubit*, respectivamente. La presencia en esos planos hace posible detectar el error mediante mediciones. La manera de generalizar esta idea consiste en considerar que el espacio H_e puede descomponerse en subespacios que caracterizan los errores identificables. Definimos a continuación estos subespacios.

Definición 9.7. Dado un espacio de codificación H_e y un error consistente en un endomorfismo unitario E, llamamos **subespacio del síndrome de error** E a un subespacio H_{S_E} de H_e tal que para todo estado inicial $|\psi_0\rangle_e$, su estado final $|\psi_1\rangle_e = E|\psi_0\rangle_e$ está en este subespacio H_{S_E}.

Detengámonos ahora a considerar algunas propiedades que han de tener los subespacios de síndromes. Por un lado, se trata efectivamente de un subespacio de Hilbert del espacio H_e y que tiene la misma dimensión que el código H_c. Podemos expresar este sencillo resultado como una proposición:

Proposición 9.2. *Dado un endomorfismo unitario ("el error") E en el espacio de codificación H_e, el subespacio de síndrome $H_{S_E} \subset H_e$ es un subespacio de Hilbert de H_e y su dimensión es la misma que la del subespacio código H_c.*

Demostración. Si la dimensión del subespacio H_c es d, existirá una base ortonormal de H_c $\{|e_i\rangle\}_{i=1,\dots,d}$. Al ser E una transformación unitaria, conserva los productos internos, por lo que $\{E|e_i\rangle\}_{i=1,\dots,d}$ serán también d vectores ortonormales de H_e, que forman una base ortonormal para H_{S_E}. □

Por otra parte, el caso de ausencia de error corresponde al caso trivial en el que la aplicación unitaria es la identidad. En tal situación, el subespacio de síndrome coincide de manera obvia con el código: $H_{S_I} = H_c$.

Además, si tenemos dos tipos de errores diferentes dados por endomorfismos unitarios E y E' que requieren correcciones diferentes, entonces los subespacios de síndrome correspondientes a ambos han de ser ortogonales. Y, por la proposición anterior, ambos tendrán la misma dimensión d.

Llegamos pues a la idea de que para un espacio de codificación H_e los subespacios de síndrome han de tener todos la misma dimensión y ser ortogonales. Esto implica que si el código corrector H_c tiene dimensión d_c y el espacio de codificación tiene dimensión d_e, entonces el número de errores diferentes que se puede corregir será un número k tal que $d_e \geqslant d_c \times k$. De hecho, podemos considerar que se cumple la igualdad, ya que si hubiera subespacios ortogonales adicionales a los que puede llegar la evolución temporal de H_e, estos pueden ser añadidos como un tipo de error adicional y si hubiera subespacios inalcanzables para la evolución temporal de H_e, podemos considerar un espacio de errores más reducido con las dimensiones

estrictamente necesarias para el proceso de corrección. De esta manera, llegamos a la siguiente conclusión en forma de proposición:

Proposición 9.3. *Dado un espacio original H_0 de dimensión d_0, un espacio de codificación H_e de dimensión d_e y un código corrector H_c, cuya dimensión ha de coincidir con la d_0 del espacio original, entonces d_e ha de ser múltiplo de d_0 y el espacio de codificación permitirá corregir un número $d_e/d_0 - 1$ de endomorfismos unitarios diferentes.*

El ejemplo de la inversión de bit que hemos utilizado como modelo para construir este formalismo abstracto cumple claramente esta condición: el espacio original de un solo *qubit* tiene dimensión 2. Y al extender el sistema con dos *qubits* ancilares para tener un sistema de tres *qubits*, obtenemos un espacio de codificación de dimensión 8. Podemos, por consiguiente, descomponer el espacio de codificación en cuatro subespacios de dimensión 2. Uno de estos será el código corrector H_c y los otros tres serán los tres subespacios de síndrome correspondientes a los errores dados por los endomorfismos unitarios $\sigma_x \otimes I \otimes I$, $I \otimes \sigma_x \otimes I$ y $I \otimes I \otimes \sigma_x$.

Siguiendo la analogía con el ejemplo de la inversión de bit, definimos ahora el concepto de detección del síndrome:

Definición 9.8. Dados un espacio de codificación H_e, un error consistente en un endomorfismo unitario E y su espacio de síndrome correspondiente H_{S_E}, se llama **comprobación del síndrome de error** E a la aplicación en H_e de proyección sobre el subespacio H_{S_E}.

Definición 9.9. Se llama **detección del síndrome** al cálculo de los valores esperados de las comprobaciones de síndrome de error para todos los subespacios de síndrome, incluido el espacio del código corrector (ausencia de error) para encontrar cuál de esos valores esperados es el único no nulo. De esta manera, se determina cuál de los errores es más probable que haya ocurrido.

Una vez detectado el síndrome habrá que devolver el estado al subespacio del código corrector H_c para eliminar el error. Formalmente, este procedimiento consiste en aprovechar los isomorfismos que existen entre los subespacios de síndromes y el código corrector. Podemos dar forma a esta idea con una nueva definición:

Definición 9.10. Dado que para un endomorfismo unitario E el subespacio de síndrome del error E, H_{S_E}, tiene la misma dimensión que el código corrector H_c, existirá un isomorfismo entre ambos subespacios $R : H_{S_E} \to H_c$. A este isomorfismo se le llama **recuperación del error**.

Para completar la terminología que nos resultará útil, daremos un nombre a la combinación de la detección del síndrome y la recuperación:

Definición 9.11. A la detección del síndrome seguida de la aplicación de recuperación se la llama **corrección del error por detección y recuperación**.

En realidad, es habitual que estas dos partes de detección y recuperación puedan combinarse en una sola transformación. Esto lo hemos visto en el caso de la inversión de bit. De manera más general, llamaremos corrección a una aplicación lineal que nos devuelva desde cualquiera de los subespacios de síndrome al código corrector:

Definición 9.12. Se llama **corrección del error** a una aplicación unitaria del espacio de codificación H_e en el código corrector H_c. La "corrección del error por detección y recuperación" de la definición anterior es un caso particular de "corrección del error".

Armados con las transformaciones y espacios que hemos definido, ya tenemos todo el proceso desde la configuración inicial del sistema de *qubits* hasta la corrección final. Tan solo nos resta volver al espacio original, para lo que necesitaremos una última aplicación lineal:

Definición 9.13. Se llama **decodificación** a la aplicación lineal inversa de la codificación $C^{-1} : H_c \to H_0$.

El proceso de corrección de errores consiste entonces en aplicar una codificación, dejar que el sistema evolucione en el espacio de errores, aplicar una corrección y, finalmente, la decodificación que devuelve el sistema considerado al espacio de Hilbert original.

Un resultado muy importante es el del siguiente teorema, gracias al cual sabemos que basta con que un código permita corregir un conjunto reducido de errores para que se pueda corregir cualquier error.

Teorema 9.4. *Si un código corrector permite corregir dos errores, correspondientes a dos endomorfismos unitarios E_1 y E_2, entonces también permite corregir cualquier combinación lineal de E_1 y E_2 [13, p. 4].*

Demostración. Es consecuencia inmediata de la linealidad de las aplicaciones unitarias. Si dados un espacio original H_0, un espacio de codificación H_e y un código corrector H_c, existe una corrección de error $C_1 : H_e \to H_c$ para E_1 y otra corrección de error $C_2 : H_e \to H_c$ para E_2, entonces todo error combinación lineal de E_1 y E_2 tendrá La forma $E = a_1E_1 + a_2E_2$, donde $a_1, a_2 \in \mathbb{C}$. Tomando entonces una corrección de error $c = a_1C_1 + a_2C_2$ se consigue corregir E. □

Gracias a este teorema (9.4), dado un sistema de n *qubits* bastará con probar que un código corrector permite corregir los errores correspondientes al grupo de Pauli $\mathcal{P}_n$ para poder afirmar que el código permite corregir cualquier error.

9.3 Condiciones de corrección de errores cuánticos

En esta sección, vamos a formular el teorema más importante de la teoría de códigos correctores cuánticos, que nos da una caracterización explícita para aquellos subespacios de Hilbert que son códigos correctores. Las condiciones que ha de cumplir un subespacio de Hilbert para ser código corrector surgen de manera natural al examinar el comportamiento necesario para que los errores se manifiesten de tal manera que se puedan identificar y corregir.

Siguiendo la notación de la sección anterior, sea H_0 el espacio de Hilbert de un sistema de n *qubits*. H_0 tiene pues dimensión 2^n y tendrá una base ortonormal $\{|i\rangle\}_{i=0,\dots,2^n-1}\}$. Sea H_e el espacio de codificación, que será un sistema de m *qubits* con dimensión 2^m, siendo $m > n$. En este espacio de codificación puede producirse un error E que ha de ser una combinación lineal de endomorfismos unitarios del grupo de Pauli $\mathcal{P}_m$. Supondremos que los errores que se han de corregir son un subconjunto de la totalidad de aplicaciones del grupo de Pauli, por lo que los posibles errores formarán un espacio vectorial $\mathcal{E}$ de endomorfismos unitarios en el que existe una base ortonormal $\{E_i\}_{i=1,\dots,k}$ en donde $k \leqslant 2^{2m}$ es la dimensión de este espacio de errores y cada E_i es una aplicación del grupo de Pauli $\mathcal{P}_n$. La identidad I ha de ser uno de los E_i, pues siempre se incluye como caso trivial la posibilidad de que no se manifieste ningún error. Así, cualquier error sobrevenido en el espacio H_e será un endomorfismo unitario $E = \sum_{i=1}^{k} \lambda_i E_i$. Asumimos, además, que existe un código corrector H_c que permite corregir estos errores. Tal como hemos visto en la sección anterior, H_c ha de tener la misma dimensión, 2^n, que el espacio original H_0, luego tendrá una base ortonormal $\{|\bar{0}\rangle, \dots \left|\overline{2^n - 1}\right\rangle\}$ a cuyos elementos llamamos las palabras del código.

Bajo las hipótesis del párrafo anterior, ¿que condiciones deberá cumplir H_c para que permita corregir los errores del espacio $\mathcal{E}$? En primer lugar, la condición más evidente consiste en que los errores E_a $(0 \leqslant a \leqslant k)$ nunca puedan convertir unas palabras de código en otras, pues en tal caso sería imposible recuperar la palabra de código original. Así pues, dada una palabra de código $|\bar{i}\rangle$ $(0 \leqslant i \leqslant 2^n - 1)$ de H_c, el error E_a la convertirá en un vector $E_a|\bar{i}\rangle$ que ha de ser ortogonal no solo a cualquier otra palabra de código $|\bar{j}\rangle$ con $j \neq i$ $(0 \leqslant j \leqslant 2^n - 1)$, sino también a sus imágenes por cualquier error $E_b|\bar{j}\rangle$ $(0 \leqslant b \leqslant k)$ para eliminar cualquier ambigüedad respecto a la palabra de código original. Esto implica que para todas las combinaciones de índices a, b, i y j con $i \neq j$ ha de cumplirse la condición de ortogonalidad $\langle\bar{i}|E_a^\dagger E_b|\bar{j}\rangle = 0$.

Más sutil es la situación con las imágenes por los diferentes errores de una misma palabra de código. Dada una palabra $|\bar{i}\rangle$, ¿qué relacion habrán de cumplir las imágenes por errores diferentes $E_a|\bar{i}\rangle$ y $E_b|\bar{i}\rangle$ $(0 \leqslant a, b \leqslant k,\ a \neq b)$? El caso más sencillo sería aquel en el que estas imágenes por errores diferentes son ortogonales, pues entonces se podrá determinar con exactitud el error acaecido. La condición sería $\langle\bar{i}|E_a^\dagger E_a|\bar{i}\rangle = \langle\bar{i}|E_b^\dagger E_b|\bar{j}\rangle = 0$. No obstante, esa condición es suficiente, pero no necesaria [13, p. 5], pues puede haber errores que se confundan en su manifestación sobre el estado pero que se puedan corregir a través de una detección de síndro-

me común[6]. Sin embargo, si se impone simplemente la igualdad, no necesariamente a cero, $\langle\bar{i}|E_a^\dagger E_a|\bar{i}\rangle = \langle\bar{i}|E_b^\dagger E_b|\bar{j}\rangle$, la condición sí es suficiente. La demostración, no trivial, de este último resultado puede encontrarse en las referencias [14, p. 14].

Apoyándonos en los razonamientos anteriores, estamos ya en condiciones de enunciar el teorema fundamental de los códigos correctores:

Teorema 9.5. *Sea H_0 un espacio de Hilbert de dimensión 2^n y sea H_e un espacio de Hilbert de dimensión 2^m, siendo $m > n$. Sea $\mathcal{E}$ un espacio vectorial de endomorfismos unitarios en H_e, que tiene como base ortonormal un conjunto $\{E_i\}_{i=1,\dots,k}$ de endomorfismos del grupo de Pauli $\mathcal{P}_m$ incluyendo al menos la identidad. Entonces un subespacio de Hilbert H_c de H_e de dimensión 2^n con una base ortonormal $\{|\bar{i}\rangle\}_{i=0,\dots,2^n-1}$ es un código corrector con espacio original H_0 y espacio de codificación H_e para el conjunto de errores $\mathcal{E}$ si y solo si se cumple:*

$$\langle\bar{i}|E_a^\dagger E_b|\bar{j}\rangle = C_{ab}\delta_{ij} \tag{9.10}$$

C_{ab} es una matriz cuadrada $k \times k$ hermítica, independiente de los índices i y j.

Estas condiciones nos dan un criterio para comprobar si una determinada elección de subespacio de Hilbert constituye un código corrector para un conjunto de errores dado.

10 Más códigos correctores cuánticos

En la sección sobre la inversión de bit en un solo *qubit*, vimos cómo construir un código corrector para ese tipo de error. Vamos a ver ahora otros dos códigos que nos permitirán corregir cualquier error que pueda afectar al sistema de un único *qubit*.

10.1 El código corrector de inversiones de signo

Como hemos visto, todo operador unitario que actúe sobre un sistema de un *qubit* será una combinación lineal de los operadores de Pauli σ_x, σ_y y σ_z, más la identidad. Hemos introducido ya un código corrector que reduce la aparición de errores de tipo σ_x. Vamos a estudiar ahora cómo se pueden corregir los demás tipos de errores que, en realidad, veremos que comprenden solamente un tipo adicional: la inversión de signo de una de las componentes del estado (también llamada inversión de fase, *phase flip* en inglés, pues equivale a multiplicar por $e^{i\pi}$ una de las componentes).

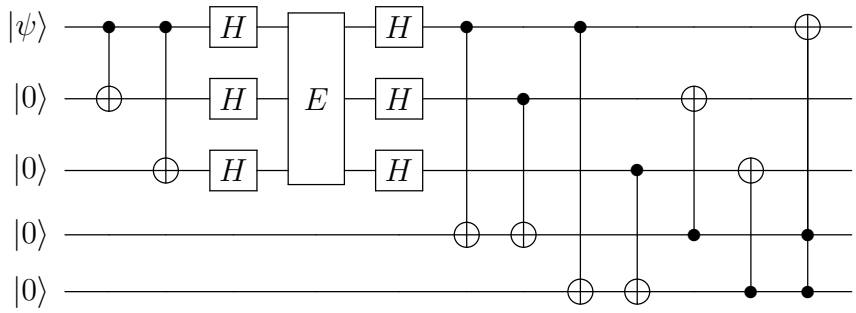
Dado un sistema de un solo *qubit*, cuyo estado viene descrito por un vector $|\psi\rangle =$

[6]El código de Shor de nueve *qubits*, que veremos más adelante es un caso de código en el que se da esta situación.

$a|0\rangle + b|1\rangle$, con $a, b \in \mathbb{C}$ tales que $|a|^2 + |b|^2 = 1$, el operador de Pauli σ_z transforma este estado en $\sigma_z|\psi\rangle = a|0\rangle - b|1\rangle$. Se trata de una transformación unitaria que podría manifestarse de manera indeseada, al igual que pasaba con las transformaciones σ_x de inversión de bits.

El tratamiento de este nuevo tipo de error resulta extremadamente sencillo gracias a una característica llamativa de los dos tipos de errores: las inversiones de bit σ_x en la base $\{|0\rangle, |1\rangle\}$ se transforman en inversiones de signo σ_z en la base $\{|+\rangle, |-\rangle\}$ y viceversa. Este sorprendente carácter dual de los dos tipos de error hace que el planteamiento que habíamos seguido para la corrección de la inversión de bit sea aplicable también a los errores provocados por σ_z. Bastará con pasar a la base $\{|+\rangle, |-\rangle\}$ los tres *qubits* tras aplicar el código de repetición, de modo que en lugar de tener un estado $a|000\rangle + b|111\rangle$ tendremos un estado $a'|+++\rangle + b'|---\rangle$. Este cambio de base equivale a aplicar una transformación de Hadamard a los *qubits*. Tras la corrección habrá que aplicar de nuevo la transformación de Hadamard (que coincide con su inversa) para volver a la base $\{|0\rangle, |1\rangle\}$.

El circuito corrector será por consiguiente el que se muestra en la siguiente figura:



(10.1)

Hasta aquí hemos encontrado dos códigos correctores para los errores provocados por los operadores de Pauli σ_x y σ_z. Nos falta σ_y, pero en realidad este operador cumple la relación $\sigma_y = i\sigma_x\sigma_z$ por lo que equivale a la aplicación sucesiva de los dos tipos de inversión vistos, que ya sabemos cómo corregir (la multiplicación por i añade simplemente un factor complejo sin significado físico, como se indicó en una sección anterior).

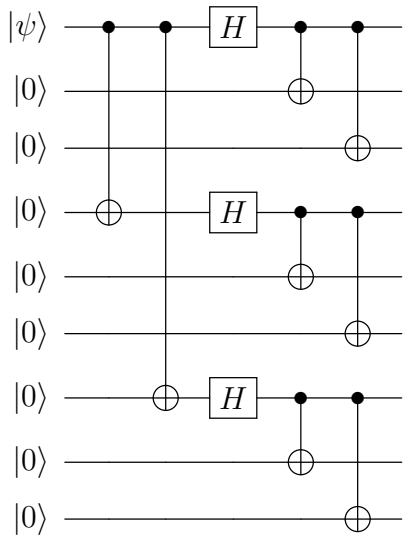
De esta manera, disponemos ya de dos códigos correctores que corrigen todos los errores posibles sobre un sistema de un *qubit* suficientemente aislado de su entorno.

10.2 El código de Shor o código de nueve *qubits*

Siguiendo la exposición de Chuang y Nielsen [21], los códigos de corrección de inversión de bits y de signo pueden combinarse en un solo código. El código resultante implica nueve *qubits* y se lo conoce como "código de Shor".

Para construir el código de Shor, se codifica primero el *qubit* original mediante el

código de inversión de signo, lo cual implica un sistema de tres *qubits*. A continuación se codifica cada uno de estos tres *qubits* mediante el código de inversión de bits, con lo que tenemos un sistema de nueve *qubits*. La fase de codificación corresponderá a la siguiente puerta lógica:



(10.2)

La transformación de codificación de un vector de estado $|\psi\rangle = a|0\rangle + b|1\rangle$, con $a, b \in \mathbb{C}$ y $|a|^2 + |b|^2 = 1$ es la siguiente:

$$\begin{aligned} |\psi\rangle = a|0\rangle + b|1\rangle \mapsto \frac{a}{2\sqrt{2}}(|000\rangle + |111\rangle)^{\otimes 3} + \frac{b}{2\sqrt{2}}(|000\rangle - |111\rangle)^{\otimes 3} = \\ = \frac{a+b}{2\sqrt{2}}(|000000000\rangle + |111111000\rangle + |000111111\rangle + |111000111\rangle) + \\ + \frac{a-b}{2\sqrt{2}}(|000111000\rangle + |111000000\rangle + |000000111\rangle + |111111111\rangle) \end{aligned} \tag{10.3}$$

El espacio de codificación consiste pues en un subespacio de dimensión 8 del espacio de Hilbert de 512 dimensiones, cuyos vectores base son los triples productos tensoriales $(|000\rangle + |111\rangle)^{\otimes 3}$ y $(|000\rangle - |111\rangle)^{\otimes 3}$ (recordemos que $|000\rangle$ y $|111\rangle$ son a su vez simplemente notaciones abreviadas de los productos tensoriales $|0\rangle^{\otimes 3}$ y $|1\rangle^{\otimes 3}$).

Al combinar en un solo elemento la protección frente a inversiones de bit e inversiones de fase, esta puerta lógica elimina cualquier tipo de error que pueda afectar a un sistema de un único *qubit*.

10.3 Primera aproximación a sistemas de múltiples *qubits*

Una vez vista la manera de corregir un sistema de un único *qubit*, la primera aproximación al tratamiento de un sistema de $n > 1$ *qubits* consiste en considerar que los

qubits están suficientemente aislados entre sí como para que no haya acoplamiento entre ellos. En tal caso, el sistema de n *qubits* se comporta como un producto tensorial de los n subsistemas de cada *qubit* y se puede aplicar el producto tensorial del código de Shor en el espacio de Hilbert de 2^{9n} dimensiones.

En las secciones siguientes veremos otros códigos más generales para sistemas de múltiples *qubits*.

10.4 Los códigos correctores CSS

En la construcción del código corrector de inversiones de bit, nos habíamos basado en las ideas de un código corrector clásico, el de triple repetición, para deducir una versión cuántica. Partíamos de un código clásico en el que los bits '0' y '1' se codifican como '000' y '111', y lográbamos construir un código en el que los *qubits* $|0\rangle$ y $|1\rangle$ se codifican en un sistema de tres *qubits* en el que se toma como código corrector el subespacio de Hilbert generado por $\{|000\rangle, |111\rangle\}$. A raíz de ese éxito en la construcción de un código corrector cuántico por analogía con un código clásico, la pregunta que surge es si se podrán definir procedimientos constructivos que den lugar a códigos cuánticos a partir de códigos clásicos. Es decir, ¿hemos de conformarnos con aplicar ideas *ad hoc* tomadas de los códigos clásicos para construir códigos cuánticos? ¿O podemos ir más allá y definir procedimientos constructivos capaces de generar códigos cuánticos de manera sistemática a partir de códigos clásicos? La respuesta a esta última pregunta es afirmativa y los códigos correctores de Calderbank-Shor-Steane (en lo sucesivo, CSS) son precisamente uno de estos procedimientos constructivos con el que se obtienen códigos cuánticos a partir de códigos lineales clásicos.

Hay dos observaciones importantes que subyacen a la idea de los códigos CSS. Por un lado, vimos al desarrollar el código cuántico de corrección de inversiones de bit que este tipo de error se puede gestionar de manera similar a la corrección clásica. Por otro lado, en el segundo código de tres *qubits* que hemos introducido, el de inversiones de fase, hemos visto que bastaba con aplicar un cambio de base en el sistema, pasando de la base $\{|0\rangle, |1\rangle\}$ a la base $\{|+\rangle, |-\rangle\}$ (geométricamente, una rotación en la esfera de Bloch) para poder corregir los errores de inversión de fase ¡como si fueran inversiones de bit! Esto apunta a la idea de que podemos construir un código cuántico basándonos en uno clásico C_1 para corregir las inversiones de bit en la base $\{|0\rangle, |1\rangle\}$, transformar el sistema a la base $\{|-\rangle, |+\rangle\}$ y volver a aplicar un código para corregir inversiones de bit, el mismo C_1 o uno diferente C_2, en esta nueva base, con lo que corregiremos las inversiones de fase. Como hemos visto, la corrección de estos dos tipos de errores supone la corrección de cualquier error sobre un sistema de *qubits*. Esta idea intuitiva es la que conduce a la definición del código CSS, que es la siguiente:

Definición 10.1. Sean dos códigos lineales clásicos C_1 y C_2 de tipos (n, k_1) y (n, k_2) respectivamente y tales que $C_2 \subset C_1$. Entonces se llama **código CSS(C_1, C_2)** al

subespacio de Hilbert generado por los estados:

$$\left\{ \frac{1}{\sqrt{|C_2|}} \sum_{y \in C_2} |x + y\rangle \right\}_{x \in C_1} \tag{10.4}$$

En donde la suma $x + y$ debe interpretarse como una operación XOR (o lo que es lo mismo, suma módulo 2 bit a bit) de las palabras de los códigos clásicos C_1 y C_2. Los vectores $\frac{1}{\sqrt{|C_2|}} \sum_{y \in C_2} |x + y\rangle$ se pueden representar también como $|x + C_2\rangle$. Se puede demostrar que si dos elementos $x_1 \in C_1$ y $x_2 \in C_2$ son tales que $x_1 + x_2 \in C_2$ (o $x_1 - x_2$; recordemos que las operaciones con cadenas de bits son siempre módulo 2) entonces $|x_1 + C_2\rangle = |x_2 + C_2\rangle$. Así pues, en lenguaje de teoría de grupos habrá un $|x + C_2\rangle$ diferente para clase lateral de C_1/C_2. Y si x_1 y x_2 pertenecen a clases laterales diferentes de C_2, entonces no existe ningún par de elementos $y_1 \in C_2$, $y_2 \in C_2$ tales que $x_1 + y_1 = x_2 + y_2$. Esto implica que en este caso $|x_1 + C_2\rangle$ y $|x_2 + C_2\rangle$ son ortogonales. De este razonamiento se deduce que hay tantos vectores ortogonales $|x + C_2\rangle$ como clases laterales de C_2 en C_1. Luego la dimensión del código CSS(C_1, C_2) es igual a $|C_1|/|C_2| = 2^{k_1 - k_2}$.

Proposición 10.1. *Sean dos códigos binarios lineales clásicos C_1 y C_2 de tipos (n, k_1) y (n, k_2) respectivamente y tales que $C_2 \subset C_1$. Si tanto C_1 como $C_2^\perp$ pueden corregir t errores (o, en otras palabras, la mínima distancia de Hamming es $2t + 1$ para ambos), entonces el código $CSS(C_1, C_2)$ es un código corrector cuántico que corrige cualquier error sobre t qubits.*

Demostración. La expresión general del error sobre t *qubits* consiste en los endomorfismos del grupo de Pauli de peso t. La demostración consistirá en un desarrollo rutinario, cuyos detalles omitimos, para comprobar que se verifican las condiciones de corrección (9.10) para los vectores (10.4) tomando cualquier endomorfismo del grupo $\mathcal{P}_n$ Pauli de peso igual o menor que t. □

La definición y proposición anteriores definen una familia de códigos correctores a partir de códigos clásicos. Pero ¿cuál será el mecanismo para la recuperación del error? Apuntábamos al principio al hecho de que la corrección tanto de errores de inversión de bit como de errores de inversión de fase requiere una transformación del sistema de la base $\{|0\rangle, |1\rangle\}$ a la base $\{|+\rangle, |-\rangle\}$, lo que equivale a una rotación sobre la esfera de Bloch. Esta rotación se lleva a cabo mediante la transformación de Hadamard. Por lo tanto, la detección y corrección se podrá hacer en dos fases: primero se detectan y corrigen las inversiones de bit utilizando el código C_1 y después se aplican transformaciones de Hadamard a los *qubits* para "rotar" el sistema, y se lleva a cabo una detección y corrección totalmente análoga utilizando el código $C_2^\perp$. Acto seguido se vuelven a aplicar transformaciones de Hadamard para deshacer la rotación sobre la esfera de Bloch. Este mecanismo de detección y corrección en dos fases diferenciadas justifica el uso de dos códigos lineales clásicos C_1 y C_2. En cuanto a la detección y corrección en cada una de estas fases, el mecanismo consiste

en añadir *qubits* ancilares sobre los que se aplican transformaciones unitarias que incorporan las matrices de control de los códigos C_1 y $C_2^{\perp}$. Esto da lugar a estados en los subespacios de síndrome, sobre los que se efectúa la corrección mediante la aplicación de puertas CNOT que corrigen los *qubits* alterados. Una explicación detallada puede encontrarse en la obra de Nielsen y Chuang [21, pp. 450 y 451].

Un caso particularmente habitual de códigos CSS es aquel en que el segundo código clásico C_2 se toma como el dual del primero. El más conocido es el llamado código de Steane, que es el código CSS tomando como código clásico C_1 el código lineal de Hamming (7, 4) junto a su dual. Como el código dual del (7, 4) de Hamming es un código lineal de tipo (7, 3) y la mínima distancia de Hamming para estos códigos es 3, el código de Steane será un código con parámetros $n = 7$, $k_1 - k_2 = 1$ y $t = 1$; esto es, un código cuántico que codifica un *qubit* en siete *qubits* y que resuelve cualquier error sobre ese único *qubit* original.

Al tratarse de un código para un *qubit*, la fórmula (10.4) deberá darnos dos elementos $|\bar{0}\rangle$ y $|\bar{1}\rangle$. Para calcularlos, veamos primero cuáles son las palabras del código lineal de Hamming (7, 4), al que llamaremos simplemente C. Estas palabras se obtienen mediante aplicación directa de la matriz generatriz (7.5) a las dieciséis cadenas binarias de longitud 4. De esa manera, se obtienen las dieciséis palabras de código siguientes:

$$\begin{aligned} C = \{&0000000, 1110000, 1001100, 0101010, \\ &1101001, 0111100, 1011010, 0011001, \\ &0100101, 1000011, 1100110, 0001111, \\ &0110011, 1010101, 0010110, 1111111\} \end{aligned} \tag{10.5}$$

Análogamente, aplicando la transpuesta de la matriz de control de (7.5), que es matriz generatriz para el código dual $C^{\perp}$, a las ocho cadenas binarias de longitud 3, se obtienen las palabras del código dual (7, 3), que son las siguientes:

$$\begin{aligned} C^{\perp} = \{&0000000, 1010101, 0110011, 0001111, \\ &0111100, 1011010, 1100110, 1101001\} \end{aligned} \tag{10.6}$$

Se observa que $C^{\perp} \subset C$, por lo que de acuerdo con la proposición (10.1) se puede construir un código CSS con el par de códigos clásicos C y $C^{\perp}$.

Si nos ponemos a sumar entre sí pares de las dieciséis palabras de (10.5), nos encontraremos con dos conjuntos de ocho palabras tales que cualquier par de palabras de un mismo grupo sumadas entre sí dan un elemento de (10.6), mientras que dos palabras tomadas de cada uno de los dos grupos no dan un elemento de (10.5) al ser sumadas. Estos dos conjuntos son las clases laterales del grupo cociente $C/C^{\perp}$. Por ejemplo, $0000000 + 1010101 = 1010101$, que está entre las palabras de $C^{\perp}$, por lo que 0000000 y 1010101 pertenecen a la misma clase lateral. Análogamente,

$1111111 + 1110000 = 0001111$, por lo que el par de 1111111 y 1110000 también comparten clase lateral, pero diferente de la de 0000000 y 1010101, pues si sumamos $0000000 + 1111111$ o $0000000 + 1110000$ o $1010101 + 1111111$ o $1010101 + 1110000$, el resultado no se encuentra entre las palabras de $C^{\perp}$. Por lo tanto, para construir las dos palabras $|\bar{0}\rangle$ y $|\bar{1}\rangle$ del código cuántico de Steane, basta con elegir dos representantes de las dos clases laterales, por ejemplo 0000000 y 1111111, y sumar a cada uno las ocho palabras de $C^{\perp}$ dividiendo por el factor de normalización $\sqrt{|C^{\perp}|}$ que en este caso es $\sqrt{8}$. Mediante este procedimiento, encontramos las dos palabras del código cuántico de Steane:

$$\begin{aligned}
|\bar{0}\rangle = \frac{1}{\sqrt{8}}(&|0000000\rangle + |1010101\rangle + |0110011\rangle + |0001111\rangle + \\
&+|0111100\rangle + |1011010\rangle + |1100110\rangle + |1101001\rangle) \\
|\bar{1}\rangle = \frac{1}{\sqrt{8}}(&|1111111\rangle + |0101010\rangle + |1001100\rangle + |1110000\rangle + \\
&+|1000011\rangle + |0100101\rangle + |0011001\rangle + |0010110\rangle)
\end{aligned} \tag{10.7}$$

Así pues, con el código de Steane basado en la construcción CSS hemos encontrado un nuevo código corrector para cualquier error sobre un sistema de un único *qubit*. Esto es lo mismo que habíamos logrado con el código de Shor para nueve *qubits*, pero ahora hemos sido capaces de reducir el espacio de codificación a siete *qubits*. En la sección siguiente, veremos que un formalismo más sofisticado, el de estabilizadores, nos permitirá encontrar un código corrector para el sistema de un *qubit* aun más compacto, de solo cinco *qubits*.

10.5 El formalismo de los estabilizadores

El formalismo de estabilizadores, debido a Daniel Gottesman [14], proporciona una manera sistemática de construir nuevos códigos correctores. Los códigos que hemos visto hasta ahora, como el de nueve *qubits* de Shor o los códigos CSS, aparecen dentro de este formalismo como casos particulares, con lo que se consigue un marco teórico mucho más potente que amplía las ideas que hemos introducido en las secciones anteriores.

Antes de pasar a las definiciones formales, vamos a ver a partir del ejemplo más sencillo la idea sobre la cual Gottesman construyó la teoría de los estabilizadores. En el caso del código corrector de tres *qubits* para los errores de inversión de bit en un solo *qubit*, vimos que el código permitía corregir inversiones en uno de los *qubits*, lo que corresponde a las aplicaciones lineales resultado de hacer el producto tensorial de la aplicación de Pauli X en uno de los *qubits* con la identidad para los otros dos *qubits*. Es decir, el código corrige las aplicaciones $X \otimes I \otimes I$, $I \otimes X \otimes I$ y $I \otimes I \otimes X$.

Aquí resulta conveniente introducir un tipo de notación más compacta para estos productos tensoriales, a los que nos referiremos en lo sucesivo como X_1, X_2 y X_3,

respectivamente. Es decir, emplearemos un subíndice para el *qubit* al que se aplica la transformación. Si, por ejemplo, tenemos un producto tensorial como $X \otimes Y \otimes Z$, escribiremos entonces $X_1 Y_2 Z_3$. La ausencia de un índice concreto corresponderá a la presencia de la identidad para ese *qubit*, de modo que para $Z \otimes I \otimes X$ escribiremos $Z_1 X_3$. Esto justifica el uso de X_1, X_2 y X_3 para los tres errores de inversión de bit sobre un único *qubit*.

La idea de los estabilizadores parte de una observación importante: aparte de los errores que detecta y corrige el código corrector de tres *qubits*, hay también errores sobre más de un *qubit* que actúan sobre el código como si se tratara de la identidad. Un ejemplo es $Z_1 Z_2$, la doble inversión de fase en el primer y el segundo *qubit*. En efecto, si tomamos la base $\{|000\rangle, |001\rangle, |010\rangle, |011\rangle, |100\rangle, |101\rangle, |110\rangle, |111\rangle\}$, los vectores base del código corrector $|000\rangle$ y $|111\rangle$ tendrán coordenadas $(1,0,0,0,0,0,0,0)$ y $(0,0,0,0,0,0,0,1)$, respectivamente, y la matriz coordenada de $Z_1 Z_2$ calculada mediante productos de Kronecker será la siguiente:

$$Z_1 Z_2 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \otimes \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \otimes \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} =$$

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{10.8}$$

Mediante esta representación en coordenadas se comprueba que los vectores (1, 0, 0, 0, 0, 0, 0, 0) y (0, 0, 0, 0, 0, 0, 0, 1), que son las palabras del código corrector de tres *qubits* para la inversión de bit, son invariables para esta transformación. Lo mismo ocurriría con $Z_1 Z_3$ y con $Z_2 Z_3$. Hemos encontrado así varios endomorfismos unitarios del grupo de Pauli $\mathcal{P}_3$ que tienen el efecto de dejar impertérritos a los vectores del subespacio de Hilbert del código. Adelantándonos a las definiciones formales, diremos que estas aplicaciones "estabilizan" al código corrector y las llamaremos "estabilizadores". En lenguaje más matemático, lo que hemos encontrado son endomorfismos unitarios definidos en el espacio de codificación de tres *qubits* para los cuales el código corrector es un autoespacio de dimensión 2 con autovalor 1. Es decir, dado el espacio de codificación de dimensión 8 para tres *qubits*, el código corrector para la inversión de bit es el subespacio de Hilbert formado por aquellos vectores que cumplen $Z_1 Z_2 |\psi\rangle = |\psi\rangle$ y $Z_1 Z_3 |\psi\rangle = |\psi\rangle$. También cumplen $Z_2 Z_3 |\psi\rangle = |\psi\rangle$, pero esta condición es redundante ya que se cumple $Z_2 Z_3 = (Z_1 Z_2)(Z_1 Z_3)$. Además, hemos de recordar la propiedad (2.10) que cumplen las aplicaciones del grupo $\mathcal{P}_n$ de Pauli, la de que su cuadrado tiene que valer I o $-I$. Evidentemente, en este caso se cumple $(Z_1 Z_2)^2 = (Z_1 Z_3)^2 = I$. En lenguaje de teoría de grupos, los cua-

tro endomorfismos unitarios $\{I, Z_1Z_2, Z_1Z_3, Z_2Z_3\}$ forman un subgrupo del grupo $\mathcal{P}_3$ de Pauli, el llamado subgrupo de Klein, que tiene como generadores a Z_1Z_2 y Z_1Z_3. Este subgrupo de estabilizadores nos da una caracterización del subespacio del código corrector.

Pero ¿qué nos aporta esto para corregir errores? ¿Cómo nos sirven estas operaciones Z_1Z_2 y Z_1Z_3 para detectar y corregir los errores X_1, X_2 y X_3? La clave está en observar que los errores anticonmutan con los estabilizadores. Por ejemplo, $(Z_1Z_2)X_1 = -X_1(Z_1Z_2)$. Esto quiere decir que los estados de tipo $X_1|\psi\rangle$ son autovectores de autovalor -1 para el endomorfismo Z_1Z_2, lo cual proporciona el mecanismo para corregir el error: dado un vector $|\psi\rangle$ que representa un estado del sistema que puede haber sufrido un error de inversión de bit, evaluando $Z_1Z_2|\psi\rangle$ y $Z_1Z_3|\psi\rangle$ se comprueba si el vector esta en el subespacio de autovalor 1 para los dos estabilizadores o si, por el contrario, se encuentra en un subespacio de autovalor -1, lo que indicaría que se ha producido un error. Esta explicación intuitiva para el caso del código corrector más simple nos guía hacia el procedimiento general que habrá que seguir para establecer una teoría de códigos basada en estabilizadores. Para un sistema de n *qubits* se podrán estudiar diferentes subgrupos del grupo $\mathcal{P}_n$ de Pauli viendo para qué subespacios de Hilbert esos subgrupos son estabilizadores. Y esos subespacios serán códigos correctores para aquellos elementos del grupo de Pauli $\mathcal{P}_n$ que anticonmuten con los estabilizadores.

Una vez explicada la idea intuitiva vamos a pasar a construir el formalismo matemático. En primer lugar, definiremos formalmente los conceptos de "subespacio estabilizado" y "estabilizador" que hemos introducido de manera informal.

Definición 10.2. Dado el espacio de Hilbert H correspondiente a un sistema de n *qubits* y un subgrupo S del grupo $\mathcal{P}_n$ de Pauli, se llama **subespacio estabilizado por** S al conjunto $V_S := \{|\psi\rangle \in H \mid T|\psi\rangle = |\psi\rangle \ \forall T \in S\}$. Al subgrupo S se le llama el **estabilizador** de V_S.

Como consecuencia inmediata de la linealidad, todo subespacio estabilizado por un subgrupo de Pauli es en efecto un subespacio de Hilbert, lo que justifica la denominación.

Proposición 10.2. *Dado el espacio de Hilbert H correspondiente a un sistema de n qubits y un subgrupo S del grupo $\mathcal{P}_n$ de Pauli con un conjunto mínimo de r generadores* [7] *$G = \{g_1, \ldots, g_r\}$, un vector $|\psi\rangle \in H$ pertenece al subespacio estabilizado por S si y solo si $T|\psi\rangle = |\psi\rangle \ \forall g_i \in G$.*

Demostración. La naturaleza necesaria de la condición es trivial, por definición de subespacio estabilizado. En cuanto al hecho de que es también suficiente, basta con ver que si $T \in S$ no es parte del conjunto de generadores entonces se podrá expresar como resultado de una o varias operaciones de grupo entre los generadores, por lo

[7] En está definición, se asume conocido el concepto de "generadores" de un grupo, que son un conjunto de elementos del grupo tales que cualquier elemento del grupo puede obtenerse mediante la aplicación de la operación del grupo sobre ellos y sus inversos.

que tendremos $T = \prod_{i=1}^{k} g_i$, en donde cada g_i es un generador con posible repetición. Entonces se cumple $T|\psi\rangle = \prod_{i=1}^{k} g_i|\psi\rangle = |\psi\rangle$. □

Dado un subgrupo cualquiera S del grupo $\mathcal{P}_n$ de Pauli, puede ocurrir que el subespacio estabilizado por S en el sistema de n *qubits* sea el espacio vectorial trivial formado por el vector nulo. La proposición siguiente da una condición necesaria y suficiente para que un subgrupo estabilice a un subespacio no trivial.

Proposición 10.3. *Dado el espacio de Hilbert H correspondiente a un sistema de n qubits y un subgrupo S del grupo $\mathcal{P}_n$ de Pauli, el subespacio V_S estabilizado por S es un subespacio no trivial si y solo si se cumplen dos condiciones: 1) S es un subgrupo abeliano. 2) S no incluye $-I$.*

Demostración. Evidentemente, $-I$ no puede ser nunca un estabilizador y si hubiera dos elementos s_1 y s_2 de S que no conmutaran entre sí, entonces como elementos del grupo $\mathcal{P}_n$ de Pauli tendrían que anticonmutar y se tendría $s_1 s_2|\psi\rangle = -s_2 s_1|\psi\rangle$ para cualquier vector $|\psi\rangle$ del espacio de Hilbert total, pero $s_1|\phi\rangle = |\phi\rangle$ y $s_1|\phi\rangle = |\phi\rangle$ para todo $|\phi\rangle \in V_S$, lo cual es una contradicción. Luego las dos condiciones expuestas son necesarias.

La prueba formal de la suficiencia requiere más trabajo y la omitiremos. Puede encontrarse en la obra de Nielsen y Chuang [21, p. 458]. □

Una consecuencia de la particularidad del grupo $\mathcal{P}_n$ de Pauli, compuesto por productos tensoriales de las cuatro matrices de Pauli 2×2 (incluida la identidad) es el hecho de que todo subgrupo abeliano tiene orden 2^{n-k} para algún entero k tal que $0 \leqslant k \leqslant n$ y cuenta con un conjunto de $n - k$ generadores [28, p. 567]. Por consiguiente, dado un estabilizador S de un sistema de n *qubits*, cada uno de los $n - k$ generadores tendrá un autoespacio para el autovalor 1, que incluye al código corrector, y otro autoespacio para el otro autovalor posible -1 (recordemos que los endomorfismos de un grupo $\mathcal{P}_n$ de Pauli tienen siempre autovalores ± 1), que corresponde a un subespacio de síndrome en el que se puede detectar el error correspondiente a aquellos elementos del grupo $\mathcal{P}_n$ de Pauli que anticonmutan con S. Estas condiciones dan como resultado las condiciones de corrección de errores y resultan en el teorema fundamental de los códigos correctores definidos mediante estabilizadores:

Proposición 10.4. *Dado el espacio de Hilbert H correspondiente a un sistema de n qubits y un subgrupo abeliano S del grupo $\mathcal{P}_n$ de Pauli que no contiene a $-I$, entonces S es un estabilizador no trivial del espacio H con un conjunto mínimo de generadores $G = \{g_i\}_{i=1,\ldots,k}$ y el subespacio de H estabilizado por V_S, H_S, es un código corrector que corrige un conjunto de endomorfismos $\mathcal{E}$ del grupo $\mathcal{P}_n$ de Pauli tales que todos los productos $EF^\dagger$ con $E, F \in \mathcal{E}$ o bien anticonmutan con algún generador g_k o bien pertenecen al estabilizador S [3].*

Demostración. La primera parte simplemente repite la proposición anterior (10.3).

Tenemos que probar que el estabilizador S corrige los errores que cumplen las condiciones mencionadas.

Sean dos endomorfismos (errores) $E, F \in \mathcal{E}$ tales que existe un generador $g \in G$ del estabilizador S para el que el producto $EF^\dagger$ anticonmuta. Entonces para todo vector $|\psi\rangle \in H_S$ se verifica:

$$\begin{aligned} \langle\psi|EF^\dagger|\psi\rangle = \langle\psi|EF^\dagger g|\psi\rangle = -\langle\psi|gEF^\dagger|\psi\rangle = -\langle\psi|EF^\dagger|\psi\rangle \\ \Rightarrow \langle\psi|EF^\dagger|\psi\rangle = 0 \end{aligned} \tag{10.9}$$

En particular, si tomamaos una base ortonormal $\{|i\rangle\}_{i=0,\ldots,k}$ de H_S, tendremos:

$$\langle i|EF^\dagger|j\rangle = 0 \tag{10.10}$$

En el otro caso que menciona el teorema, sean dos endomorfismos $E, F \in \mathcal{E}$ tales que el producto $EF^\dagger$ pertenece al estabilizador S. Entonces para la base ortonormal $\{|i\rangle\}_{i=0,\ldots,k}$ de H_S tenemos:

$$\langle i|EF^\dagger|j\rangle = \langle i|j\rangle \tag{10.11}$$

Y las ecuaciones (10.10) y (10.11) equivalen a las condiciones de corrección de errores (9.10). □

Gracias a estos resultados, dado un sistema de n *qubits* se podrán estudiar los diferentes subgrupos posibles del grupo $\mathcal{P}_n$ de Pauli que sean abelianos y no incluyan a $-I$ y ver qué errores, otro subconjunto del grupo $\mathcal{P}_n$ de Pauli, permiten corregir. Esto proporciona una técnica constructiva para escudriñar todas las posibilidades de códigos correctores para un número de *qubits* dado.

El formalismo de estabilizadores incluye como casos particulares los códigos que habíamos visto hasta ahora. Hemos mencionado ya que el código de tres *qubits* de inversión de bit es el asociado al estabilizador cuyo conjunto mínimo de generadores es $\{Z_1Z_3, Z_1Z_2\}$ y corrige los errores X_1, X_2 y X_3. Análogamente, se puede comprobar que $\{X_1X_3, X_1X_2\}$ son los generadores del estabilizador asociado al código de tres *qubits* de inversión de fase, que corrige los errores Z_1, Z_2 y Z_3.

El código de Shor de nueve *qubits* corresponde al estabilizador cuyo conjunto mínimo de generadores es el siguiente:

$$\begin{aligned} G_{Shor-9} = \{&Z_1Z_2, Z_2Z_3, Z_4Z_5, Z_5Z_6, Z_7Z_8, Z_8Z_9, \\ &X_1X_2X_3X_4X_5X_6, X_4X_5X_6X_7X_8X_9\} \end{aligned} \tag{10.12}$$

Se comprueba fácilmente que con estos ocho generadores y el conjunto de todos los errores del grupo $\mathcal{P}_n$ que afectan a un solo *qubit* se cumplen las condiciones (10.4).

En cuanto a los códigos CSS, pueden reinterpretarse mediante el formalismo de estabilizadores como códigos cuyo estabilizador tiene por generadores productos tensoriales que incluyen solamente o bien transformaciones X o bien transformaciones Z. Es decir, un código que tenga un generador como Y_1 o X_1Z_2 nunca puede ser un código CSS. Los generadores del estabilizador de un código CSS se deducen de manera inmediata a partir de las matrices de control de los códigos clásicos C_1 y $C_2^{\perp}$ con los que vimos que se construían estos códigos. En particular, cada fila de la matriz de control de C_1 representa un generador formado por el producto tensorial de las transformaciones Z_i para aquellos índices de columna i en que la matriz de control tenga un 1. Análogamente, cada fila de la matriz de control de $C_2^{\perp}$ representa un generador formado por el producto tensorial de las transformaciones X_i para aquellos índices de columna i en que la matriz de control tenga un 1[8]. Con esta regla que acabamos de describir, podemos entonces deducir los estabilizadores que dan lugar al código de siete *qubits* de Steane a partir de las matrices generatriz y de control (7.5) del código clásico (7, 4) de Hamming. Por ejemplo, la primera fila de la matriz de control H del código de Hamming es (1, 0, 1, 0, 1, 0, 1), de donde se deduce el primer generador del estabilizador para el código de Steane: $Z_1Z_3Z_5Z_7$. Aplicando esto a las demás filas de H y G^T y reduciendo casos redundantes, se obtienen seis generadores para el estabilizador del código de Steane:

$$\begin{aligned} G_{Steane-7} = \{&Z_1Z_3Z_5Z_7, Z_2Z_3Z_6Z_7, Z_4Z_5Z_6Z_7, \\ &X_1X_3X_5X_7, X_2X_3X_6X_7, X_4X_5X_6X_7\} \end{aligned} \tag{10.13}$$

Esta expresión de los generadores (10.13) del estabilizador del código es equivalente a la expresión de las palabras del código, (10.7), que vimos en la sección anterior. Con el formalismo de estabilizadores descubrimos una caracterización alternativa muy compacta y que deja entrever mejor la geometría del código.

Hasta aquí, hemos utilizado el formalismo de estabilizadores para reproducir los códigos que ya habíamos introducido previamente, pero este formalismo nos permite construir muchos más códigos. Uno de ellos, el más conocido, es el **código de cinco *qubits***[9], que como su nombre indica, codifica en cinco *qubits* un sistema de un *qubit* y se suma a los códigos de Shor de nueve *qubits* y al de Steane de siete *qubits* como códigos que permiten corregir cualquier error sobre un único *qubit*. Se puede demostrar que el código de cinco *qubits* es de hecho el más pequeño posible que corrige todos los errores en un *qubit*. Sus generadores son los siguientes:

$$G_{5-qubit} = \{X_1Z_2Z_3X_4, X_2Z_3Z_4X_5, X_1X_3Z_4Z_5, Z_1X_2X_4Z_5\} \tag{10.14}$$

[8]Esta construcción de los estabilizadores de los códigos CSS mediante matrices de control, que no hemos desarrollado, puede verse en detalle en muchas obras de referencia como la de Nielsen y Chuang [21, p. 469].

[9]Este código fue mencionado por primera vez en 1996, en un artículo [17] de varios autores.

11 Computación cuántica con tolerancia frente a fallos

En las secciones anteriores, hemos visto cómo se construyen códigos correctores para evitar que un sistema de *qubits* sufra transformaciones indeseadas. Pero esta protección frente a errores ha de combinarse con el requisito de que los sistemas de *qubits* puedan ser sometidos también a transformaciones deseadas, tanto unitarias como de medición, para poder llevar a cabo operaciones lógicas. Si se decodificaran y codificaran los *qubits* justo antes y después de someterlos a una de estas transformaciones, se dejaría abierta una ventana temporal en la que pueden manifestarse errores no detectados. La solución a este problema se estudia dentro del campo de investigación conocido como **computación con tolerancia frente a fallos** (*fault-tolerant computing*, en inglés). Este tipo de técnicas se han desarrollado de manera muy prometedora en la computación cuántica y hoy en día se sabe cómo construir versiones de circuitos cuánticos como los que hemos visto en la sección sobre puertas lógicas, pero en los que las propias operaciones de las transformaciones unitarias y mediciones se llevan a cabo en el seno de un espacio de codificación.

El estudio pormenorizado de la computación con tolerancia frente a fallos excedería el propósito de este trabajo. Mencionaremos únicamente, como ejemplo de estas técnicas, el hecho de que la teoría de códigos estabilizadores permite definir transformaciones unitarias que transforman los estados del espacio de codificación tal como ocurriría con los estados del espacio original mediante otras transformaciones unitarias, típicamente más simples. Así, por ejemplo, si sometemos al código de cinco *qubits* a una transformación $X_1X_2X_3X_4X_5$, el efecto es el mismo que si hubiéramos sometido al *qubit* no codificado a una transformación X. Análogamente, el efecto de la transformación Z sobre el *qubit* del espacio original equivale al efecto de una transformación $Z_1Z_2Z_3Z_4Z_5$ sobre el espacio de codificación de cinco *qubits*. Se dice en tal caso que $X_1X_2X_3X_4X_5$ y $Z_1Z_2Z_3Z_4Z_5$ son las operaciones lógicas $\bar{X}$ y $\bar{Z}$, respectivamente, para el código de cinco *qubits*. Estas operaciones lógicas no siempre tienen una forma tan simple. En el caso del código de Shor de nueve *qubits*, las operaciones lógicas son $\bar{X} = Z_1Z_2Z_3Z_4Z_5Z_6Z_7Z_8Z_9$ y $\bar{Z} = X_1X_2X_3X_4X_5X_6X_7X_8X_9$, al contrario de lo que se podría haber esperado.

El resultado más importante de la computación cuántica con tolerancia frente a fallos es el Teorema del umbral [13, p. 41], que enunciamos a continuación sin demostrar.

Teorema 11.1. *(Teorema del umbral) Existe un valor umbral p_t tal que si toda puerta lógica de un circuito cuántico tiene una probabilidad $p < p_t$ de sufrir errores, entonces son posibles las operaciones lógicas cuánticas arbitrariamente extensas.*

Este resultado garantiza que se pueden desarrollar circuitos cuánticos arbitrariamente complejos siempre y cuando se consiga reducir la probabilidad de error en cada puerta lógica por debajo de un valor umbral, que se ha intentado calcular de muy diversas maneras. Según Daniel Gottesman [13, p. 42], una cota inferior del

umbral bastante rigurosa es la de 10^{-3}, estimación debida a Panos Aliferis, John Preskill y el propio Gottesman [1]. El Teorema del umbral es uno de los resultados teóricos más importantes de la investigación en computación cuántica y una de las principales justificaciones teóricas para la posibilidad tecnológica de que en el futuro se puedan desarrollar ordenadores cuánticos que exploten el poder computacional de los sistemas de *qubits*.

12 Conclusión

En la sección 5 sobre los escollos tencnológicos para la realización de ordenadores cuánticos comentábamos las razones que podían llevarnos al escepticismo respecto a la posibilidad de desarrollar tecnológicamente circuitos cuánticos, entre las cuales destacaba la previsible dificultad de la corrección de errores. Sin embargo, la investigación en técnicas de corrección llevada a cabo durante los últimos veinte años ha dado lugar a una amplia teoría sobre códigos correctores y computación con tolerancia frente a fallos que parece haber despejado el camino en este terreno. Muchos escollos aparentes, como el que indicábamos en el Teorema de la imposibilidad de clonación (9.1), han sido sorteados con éxito gracias al desarrollo de estas técnicas. Así, lejos de chocar con barreras teóricas infranqueables, la investigación matemática en este campo ha sido enormemente fructífera y disponemos hoy en día de todo un arsenal de códigos correctores y de puertas lógicas con tolerancia frente a fallos.

Pese a estos éxitos teóricos, hay una realidad que invita a la cautela y es, evidentemente, el hecho de que nadie ha sido capaz de desarrollar ordenadores cuánticos con un mínimo de funcionalidad. Solamente se han conseguido aparatosos montajes de laboratorio en que, bajo muchas condiciones a veces no del todo comprendidas, se consiguen ejecutar algoritmos cuánticos sobre un conjunto muy reducido de *qubits*. Una razón a la que ya hemos aludido que explica esta dificultad es el fenómeno de la decoherencia. La naturaleza evita que los fenómenos cuánticos se manifiesten en escalas que van más allá del mundo atómico y es muy difícil mantener *qubits* entrelazados y aislados sin que cualquier mínima interacción provoque el colapso del estado. En este sentido, entender mejor la naturaleza de la decoherencia y lograr mantener los *qubits* a salvo de este fenómeno es uno de los retos pendientes para que se puedan llegar a ver ordenadores cuánticos.

Otro motivo para la cautela es el hecho de que todos los resultados teóricos como los que hemos visto a lo largo de este trabajo dependen siempre de ciertas hipótesis que pueden no ser aplicables a los dispositivos físicos tecnológicamente factibles. En un artículo reciente muy crítico con los resultados actuales de la investigación en computación cuántica, M. I. Dyakonov [11] comenta cómo se asumen por lo general hipótesis cuestionables que permiten lograr esos resultados optimistas que no acaban de encontrar reflejo en la práctica. Una de esas hipótesis, que hemos dado por válida a lo largo de este trabajo, es la idea de que se pueden preparar varios *qubits* en un mismo estado inicial de referencia al que designamos $|0\rangle$. *Si eso*

es posible, la teoría nos dice que podemos pasar los *qubits* al estado ortogonal $|1\rangle$ mediante una transformación X; o bien a un estado entrelazado al máximo mediante la transformación de Hadamard. Y que podemos montar códigos correctores como el de cinco *qubits*, el de Steane, el de Shor... Todo muy prometedor, pero que al final descansa sobre hipótesis que se asumen como razonables y que no siempre está claro que lo sean. Dyakonov comenta también que los éxitos experimentales en la factorización mediante el algoritmo de Shor se basan en una versión simplificada del algoritmo que explota el conocimiento de la solución buscada. Es decir, se ha logrado factorizar $15 = 5 \times 3$ aplicando atajos que solamente son posibles porque... ¡sabemos de antemano que $15 = 5 \times 3$! En un artículo reciente en Nature [26], se incide en este hecho y se plantea que es posible también factorizar números muy grandes explotando estas técnicas simplificadas siempre que sus factores sean conocidos de antemano para preparar adecuadamente el algoritmo. La conclusión es bastante deprimente: si no supiéramos que $15 = 5 \times 3$, el algoritmo de Shor no nos habría permitido descubrirlo a día de hoy.

La conclusión, por tanto, es que a pesar del bagaje teórico del que se dispone en la actualidad, nos encontramos aún lejos de la posibilidad tecnológica de desarrollar ordenadores cuánticos. Solamente cuando se consiga controlar el fenómeno de la decoherencia y preparar un gran número de *qubits* en estados iniciales iguales se podrá vislumbrar la posibilidad de que ordenadores de uso cotidiano aprovechen el potencial de algoritmos como los de Shor y Grover. Mientras esto no sea posible, la criptografía de clave pública actual seguirá siendo segura, al menos en lo que respecta a los ataques basados en computación cuántica.

A Apéndices

A.1 Derivación de la forma general de una matriz unitaria 2×2

En este apéndice vamos a ver un resultado que, aunque sencillo, es difícil de encontrar explícitamente en la literatura sobre aplicaciones unitarias en espacios hermíticos. Este resultado se utiliza en la sección sobre las puertas lógicas cuánticas.

Sea una matriz unitaria 2×2:

$$U = \begin{pmatrix} u_{00} & u_{01} \\ u_{10} & u_{11} \end{pmatrix} \tag{A.1}$$

Por definición, el hecho de que U sea unitaria quiere decir que ha de cumplir las dos relaciones siguientes:

$$\begin{pmatrix} u_{00} & u_{01} \\ u_{10} & u_{11} \end{pmatrix} \begin{pmatrix} \overline{u_{00}} & \overline{u_{10}} \\ \overline{u_{01}} & \overline{u_{11}} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{A.2}$$

$$\begin{pmatrix} \overline{u_{00}} & \overline{u_{10}} \\ \overline{u_{01}} & \overline{u_{11}} \end{pmatrix} \begin{pmatrix} u_{00} & u_{01} \\ u_{10} & u_{11} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{A.3}$$

Expandiendo las ecuaciones matriciales (A.2) y (A.3), obtenemos las siguientes ecuaciones:

$$\begin{aligned}
u_{00}\overline{u_{00}} + u_{01}\overline{u_{01}} &= 1 \\
u_{00}\overline{u_{10}} + u_{01}\overline{u_{11}} &= 0 \\
u_{10}\overline{u_{00}} + u_{11}\overline{u_{01}} &= 0 \\
u_{10}\overline{u_{10}} + u_{11}\overline{u_{11}} &= 1 \\
\overline{u_{00}}u_{00} + \overline{u_{10}}u_{10} &= 1 \\
\overline{u_{00}}u_{01} + \overline{u_{10}}u_{11} &= 0 \\
\overline{u_{01}}u_{00} + \overline{u_{11}}u_{10} &= 0 \\
\overline{u_{01}}u_{01} + \overline{u_{11}}u_{11} &= 1
\end{aligned} \tag{A.4}$$

Para resolver este sistema de ecuaciones, podemos representar cada número complejo u_{ij} en la llamada forma polar en función de dos valores reales $r_{ij} \geqslant 0$ y $-\pi \leqslant \theta_{ij} \leqslant \pi$. De esta manera, $u_{ij} = r_{ij}e^{i\theta_{ij}}$ y $\overline{u_{ij}} = r_{ij}e^{-i\theta_{ij}}$ y las ecuaciones (A.4) adoptan la siguiente forma:

$$\begin{aligned}
r_{00}e^{i\theta_{00}}r_{00}e^{-i\theta_{00}} + r_{01}e^{i\theta_{01}}r_{01}e^{-i\theta_{01}} &= 1 \\
r_{00}e^{i\theta_{00}}r_{10}e^{-i\theta_{10}} + r_{01}e^{i\theta_{01}}r_{11}e^{-i\theta_{11}} &= 0 \\
r_{10}e^{i\theta_{10}}r_{00}e^{-i\theta_{00}} + r_{11}e^{i\theta_{11}}r_{01}e^{-i\theta_{01}} &= 0 \\
r_{10}e^{i\theta_{10}}r_{10}e^{-i\theta_{10}} + r_{11}e^{i\theta_{11}}r_{11}e^{-i\theta_{11}} &= 1 \\
r_{00}e^{-i\theta_{00}}r_{00}e^{i\theta_{00}} + r_{10}e^{-i\theta_{10}}r_{10}e^{i\theta_{10}} &= 1 \\
r_{00}e^{-i\theta_{00}}r_{01}e^{i\theta_{01}} + r_{10}e^{-i\theta_{10}}r_{11}e^{i\theta_{11}} &= 0 \\
r_{01}e^{-i\theta_{01}}r_{00}e^{i\theta_{00}} + r_{11}e^{-i\theta_{11}}r_{10}e^{i\theta_{10}} &= 0 \\
r_{01}e^{-i\theta_{01}}r_{01}e^{i\theta_{01}} + r_{11}e^{-i\theta_{11}}r_{11}e^{i\theta_{11}} &= 1
\end{aligned} \tag{A.5}$$

Agrupando primero las igualdades a 1, tenemos las cuatro ecuaciones siguientes:

$$\begin{aligned}
r_{00}^2 + r_{01}^2 &= 1 \\
r_{10}^2 + r_{11}^2 &= 1 \\
r_{00}^2 + r_{10}^2 &= 1 \\
r_{01}^2 + r_{11}^2 &= 1
\end{aligned} \tag{A.6}$$

Si observamos que la cuarta ecuación es el resultado de sumar las dos primeras y restar la tercera, tenemos que hay una ecuación redundante, y podemos expresar todos estos valores en función de uno solo de ellos, por ejemplo r_{00}:

$$\begin{aligned} r_{01} = r_{10} = \sqrt{1 - r_{00}^2} \\ r_{11} = r_{00} \end{aligned} \tag{A.7}$$

Por otra parte, tomando en (A.5) las igualdades a cero, obtenemos:

$$\begin{aligned} r_{00}\sqrt{1 - r_{00}^2}(e^{i(\theta_{00}-\theta_{10})} + e^{i(\theta_{01}-\theta_{11})}) = 0 \\ r_{00}\sqrt{1 - r_{00}^2}(e^{i(\theta_{10}-\theta_{00})} + e^{i(\theta_{11}-\theta_{01})}) = 0 \\ r_{00}\sqrt{1 - r_{00}^2}(e^{i(\theta_{01}-\theta_{00})} + e^{i(\theta_{11}-\theta_{10})}) = 0 \\ r_{00}\sqrt{1 - r_{00}^2}(e^{i(\theta_{00}-\theta_{01})} + e^{i(\theta_{10}-\theta_{11})}) = 0 \end{aligned} \tag{A.8}$$

Las ecuaciones anteriores dan lugar a dos posibilidades: primero, si $r_{00} = 0$ o $r_{00} = 1$, entonces los valores θ pueden ser cualesquiera. La matriz unitaria tendría en ese caso una de las dos formas siguientes:

$$\begin{pmatrix} e^{i\theta_{00}} & 0 \\ 0 & e^{i\theta_{11}} \end{pmatrix} \qquad \begin{pmatrix} 0 & e^{i\theta_{01}} \\ e^{i\theta_{10}} & 0 \end{pmatrix} \tag{A.9}$$

El otro caso se da cuando se cumple $0 < r_{00} < 1$. Entonces las igualdades (A.8) se convierten en igualdades entre ángulos módulo 2π:

$$\begin{aligned} \theta_{00} - \theta_{10} - \theta_{01} + \theta_{11} = \pi \\ \theta_{10} - \theta_{00} - \theta_{11} + \theta_{01} = \pi \\ \theta_{01} - \theta_{00} - \theta_{11} + \theta_{10} = \pi \\ \theta_{00} - \theta_{01} - \theta_{10} + \theta_{11} = \pi \end{aligned} \tag{A.10}$$

Las anteriores igualdades son todas equivalentes, por lo que podemos reducirlas a una sola:

$$\theta_{00} = \pi + \theta_{10} + \theta_{01} - \theta_{11} \tag{A.11}$$

Luego, la forma general de una matriz unitaria 2×2 será:

$$\begin{pmatrix} r_{00}e^{i(\pi+\theta_{10}+\theta_{01}-\theta_{11})} & \sqrt{1 - r_{00}^2}e^{i\theta_{01}} \\ \sqrt{1 - r_{00}^2}e^{i\theta_{10}} & r_{00}e^{i\theta_{11}} \end{pmatrix} \qquad r_{00} \in [0, 1], \theta_{10}, \theta_{01}, \theta_{11} \in [-\pi, \pi] \tag{A.12}$$

Referencias

[1] Panos Aliferis, Daniel Gottesman, John Preskill. *Accuracy Threshold for Postselected Quantum Computation*. Quant. Inf. Comput. 8 (2008) 181-244. 2007. arXiv:quant-ph/0703264

[2] Dave Bacon. *The No-Cloning Theorem, Classical Teleportation and Quantum Teleportation, Superdense Coding*. CSE 599d University of Washington. `http://courses.cs.washington.edu/courses/cse599d/06wi/lecturenotes4.pdf`

[3] Dave Bacon. *Stabilizer Quantum Error Correcting Codes*. CSE 599d UNiversity of Washington. `http://courses.cs.washington.edu/courses/cse599d/06wi/lecturenotes18.pdf`

[4] *Bitcoin- Protect your privacy*. Recomendaciones de seguridad en bitcoin.org. `https://bitcoin.org/en/protect-your-privacy`

[5] *What Effects Would a Scalable Quantum Computer Have on Bitcoin?* Discusión de 2013 en bitcoin.stackexchange.com. `http://bitcoin.stackexchange.com/q/6062`

[6] Gilles Brassard, Peter Høyer, Alain Tapp. *Quantum cryptanalysis of hash and claw-free functions*. LATIN'98: Theoretical Informatics, Lecture Notes in Computer Science, pp. 163-169, Springer, 1998. `http://dx.doi.org/10.1007/BFb0054319`

[7] David Deutsch. *Quantum theory, the Church-Turing principle and the universal quantum computer*. Royal Society (London), Proceedings, Series A - Mathematical and Physical Sciences (ISSN 0080-4630), vol. 400, no. 1818, p. 97-117. 1985.

[8] David Deutsch, Richard Jozsa. *Rapid Solution of Problems by Quantum Computation*. Proceedings: Mathematical and Physical Sciences, Volume 439, Issue 1907, pp. 553-558. 1992.

[9] Ronald de Wolf. *Quantum Computing: Lecture Notes*. `http://homepages.cwi.nl/~rdewolf/qcnotes.pdf`

[10] P. A. M. Dirac. *The Principles of Quantum Mechanics*. Oxford University Press. 1930.

[11] M. I. Dyakonov. *State of the Art and Prospects for Quantum Computing*. arXiv:1212.3562v1 [quant-ph]

[12] Steven J. van Enk. *Mixed States and Pure States*. `http://pages.uoregon.edu/svanenk/solutions/Mixed_states.pdf`

[13] Daniel Gottesman. *An Introduction to Quantum Error Correction and Fault-Tolerant Quantum Computation*. Proceedings of Symposia in Applied Mathema-

tics. 2009. arXiv:0904.2557v1 [quant-ph]

[14] Daniel Gottesman. *Stabilizer Codes and Quantum Error Correction*. Caltech Ph.D. Thesis. 1997. arXiv:9705052 [quant-ph]

[15] Lov K. Grover. *A Fast Quantum Mechanical Algorithm for Database Search*. Proceedings, 28th Annual ACM Symposium on the Theory of Computing (STOC), May 1996, pp. 212-219. arXiv:quant-ph/9605043

[16] Karl Kraus. *General State Changes in Quantum Theory*. Annals of Physics, Volume 64, Issue 2, pp. 311-335. 1971.

[17] Raymond Laflamme, Cesar Miquel, Juan Pablo Paz, Wojciech Hubert Zurek. *Perfect Quantum Error Correction Code*. 1996. arXiv:/9602019 [quant-ph]

[18] Carlos Munuera Gómez, Juan Gabriel Tena Ayuso. *Codificación de la información*. Universidad de Valladolid. 1997.

[19] Ashok Muthukrishnan. *Classical and Quantum Logic Gates: An Introduction to Quantum Computing*. `http://www.optics.rochester.edu/~stroud/presentations/muthukrishnan991/LogicGates.pdf`

[20] John von Neumann. *Mathematische Grundlagen der Quantenmechanik*. J. Springer, Berlin. 1932 (última edición de Springer Verlag de 1996).

[21] Michael A. Nielsen, Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 10th Anniversary Edition. 2010.

[22] Steven Roman. *Advanced Linear Algebra*. Springer, Second Edition, 2005.

[23] Peter W. Shor. *Algorithms for Quantum Computation: Discrete Logarithms and Factoring*. Proceedings, 35th Annual Symposium on Foundations of Computer Science, Santa Fe, NM, November 20–22, 1994, IEEE Computer Society Press, pp. 124–134.

[24] Peter W. Shor. *Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer*. SIAM J.Sci.Statist.Comput. 26 (1997) 1484, 1995. arXiv:quant-ph/9508027

[25] Erwin Schrödinger. *Quantisierung als Eigenwertproblem (Erste Mitteilung)*. Annalen der Physik 79 p. 361. 1926.

[26] John A. Smolin, Graeme Smith, Alexander Vargo. *Oversimplifying Quantum Factoring*. Nature Vol. 499, Macmillan Publishers Limited. 11 de julio de 2013.

[27] Umesh V. Vazirani. *CS-191x Quantum Mechanics and Quantum Computation*. CDX Berkeley online course. `https://courses.edx.org/courses/BerkeleyX/CS-191x/2013_August/`

[28] Mark M. Wilde. *Quantum Information Theory*. Cambridge University Press. 2013.