Universitat Jaume I

Escuela Superior de Tecnología y Ciencias Experimentales

Departamento de Matemáticas
Trabajo de Fin de Máster

Códigos: clásicos y cuánticos

Autor:
Laura Clara
Jover Galtier

Supervisor:
Dr. Fernando Javier
Hernando Carrillo

Índice general

Capítulo 1

Introducción

En este trabajo vamos a estudiar los códigos de corrección de errores, tanto clásicos como cuánticos. Estos códigos sirven para corregir la información digital cuando ésta se corrompe al ser enviada a través de un canal con ruido.

La teoría clásica de la información surgió a finales de la Segunda Guerra Mundial, y fue iniciada en 1948 por Claude Shannon, con su publicación "A Mathematical Theory of Communication" [2]. Durante esa época se buscó una manera más eficiente de emplear los canales de comunicación, teniendo como objetivo el encontrar una transmisión óptima de los mensajes. Ya desde la década de 1910 se comenzó a estudiar el problema, con los artículos publicados por A. Markovi. R. Hartley [3] continuó su trabajo, y fue quién desarrolló el lenguaje binario en 1927. Tras esto, en 1936, A. Turing ideó una máquina capaz de tratar información con emisión de símbolos. Y, finalmente, C. Shannon junto a W. Weaver participó en la culminación y asentamiento de la Teoría de la Información y la Computación Clásicas.

Weaver alcanzó un planteamiento superior al inicial, creando un modelo lineal simple: la información parte de una fuente, es codificada, se transmite por un canal, se decodifica y alcanza su destino. La Teoría de la Información abarca múltiples formas de transmisión y almacenamiento de información, como pueden ser la televisión, los impulsos eléctricos de los ordenadores, los sistemas de comunicación por radio, la grabación óptica de datos e imágenes...

El objetivo de la Teoría de la Información es garantizar que el transporte masivo de datos no implique una pérdida de calidad, incluso en el caso de que se produzca una compresión para su mejor transmisión o almacenamiento. De manera ideal, los datos se podrían devolver a su forma original al llegar a su destino, aunque vamos a ver que lo que podemos hacer es disminuir la probabilidad de que se produzcan errores, sin poder garantizar que el resultado sea en todos los casos la recuperación completa de la información transmitida.

Por otra parte, la computación cuántica es un modelo de computación distinto a la computación clásica. Emplea qubits en lugar de bits para almacenar la información, y precisa de nuevas puertas lógicas, permitiendo crear nuevos algoritmos.

Una misma tarea puede tener distinta complejidad en cada tipo de computación. Así, algunos problemas sencillos en la computación clásica se vuelven más complejos en la cuántica. Sin embargo, existen procesos inabarcables para la computación clásica que se podrían resolver mediante la computación cuántica.

Con el transcurso de los años, los elementos de un computador clásico se han ido volviendo más pequeños. Cada vez caben más transistores en menos espacio, por lo que los microchips se vuelven más pequeños y los chips alcanzan velocidades mayores. Sin embargo, este proceso de disminución del tamaño de los elementos de las computadoras tiene un límite, puesto que al alcanzar un tamaño extremadamente pequeño empezamos a encontrar fenómenos inesperados, como el hecho de que los electrones saltan entre los distintos canales por los que deben circular debido al efecto túnel. Esto sucede porque pasamos del mundo macroscópico al microscópico, por lo que encontramos fenómenos de la Física Cuántica que no habíamos presenciado antes en la computación.

Una partícula clásica, al encontrarse con un obstáculo rebota sin atravesarlo. Sin embargo los electrones son partículas cuánticas y en ocasiones pueden comportarse como ondas, por lo que existe una probabilidad de que puedan atravesar el obstáculo si es lo suficientemente delgado. Ésto hace que los chips dejen de funcionar correctamente al llegar a un tamaño determinado.

Es por esto que la computación clásica no tardará en llegar a su límite, surgiendo así la necesidad de desarrollar nuevas tecnologías. Aquí es donde la computación cuántica se convierte en un importante objeto de estudio. La idea de la computación cuántica surge en 1981, cuando Paul Benioff expuso su teoría para aprovechar las leyes cuánticas en el entorno de la computación.

La relación entre la información cuántica y la clásica es objeto de muchos estudios actualmente. Existen muchas similitudes entre ellas, aunque también hay diferencias sustanciales entre las dos. La información clásica no puede viajar a velocidades más rápidas que la luz, mientras que la información cuántica parece poder hacerlo (aunque vamos a ser capaces de resolver esta paradoja a lo largo del trabajo). La información clásica se puede duplicar, mientras que la cuántica no (véase referencia [10]).

Ya es conocido que la información clásica se puede proteger de la degradación empleando códigos clásicos de corrección de errores. Estos códigos parecen proteger la información clásica duplicándola, por lo que, debido al teorema que presentaremos en los siguientes capítulos que indica que los bits cuánticos no pueden duplicarse, se pensaba que estas técnicas de corrección no se podían aplicar a la información cuántica. Sin embargo, recientemente se demostró que en realidad sí existen los códigos cuánticos de corrección de errores. El objetivo de este trabajo es estudiar cómo construirlos, siguiendo para ello la siguiente estructura.

En el segundo capítulo estudiaremos los códigos clásicos, comenzando con una introducción a la motivación de su desarrollo. Tras esto estudiaremos los códigos lineales clásicos, prestando especial interés a los códigos Reed-Solomon. Por último veremos la teoría de los códigos cíclicos. En particular, para cada clase de código estudiaremos sus matrices generatriz y de paridad y la manera en la que pueden corregir los errores que se producen en el mensaje.

A continuación, desarrollaremos los fundamentos de la computación cuántica, incluyendo la definición del estado de un qubit, la construcción de las puertas lógicas cuánticas a partir de las clásicas y su empleo en la fabricación de los circuitos cuánticos.

Por último, desarrollaremos una teoría para la construcción de los códigos cuánticos y su corrección de errores, empleando como base la teoría de los códigos clásicos y los fundamentos vistos en el tercer capítulo. En primer lugar, estudiaremos los tipos de errores que se pueden producir en los qubits al transmitir la información cuántica a través del canal. A continuación, veremos la teoría de corrección de errores que indica cómo recuperar el mensaje inicial. Emplearemos todo lo visto anteriormente para alcanzar nuestro objetivo, desarrollar una teoría que nos permita construir códigos cuánticos de corrección de errores.

Capítulo 2

Códigos Clásicos

Los códigos de corrección de errores clásicos tienen muchas aplicaciones tecnológicas. En especial, la teoría de los códigos lineales clásicos presenta muchas técnicas que tienen importantes implicaciones en la teoría de corrección de errores cuántica, permitiéndonos desarrollar una amplia variedad de códigos de corrección de errores cuánticos.[2][3][4][5][6][15]

2.1. La información digital

La información digital se caracteriza por presentarse en un formato discreto. Si A es un conjunto finito y A^* es el conjunto de secuencias finitas en las que se puede dividir el conjunto A, la información digital se presenta como una secuencia $m = x_1 x_2 \cdots \in A^*$.

Una vez se tiene la información en forma de secuencias, se puede manipular o transmitir. Si la información se transmite, sigue el siguiente esquema: el emisor la envía por un canal hasta el receptor. El canal por el que se envía la información puede ser espacial o temporal. Al recibir la información, el receptor no puede estar seguro de que no se hayan producido errores en algún lugar del mensaje durante la transmisión. Sin embargo, es capaz de conocer la frecuencia con la que se producen errores en el canal, por lo que puede conocer la media de errores que se pueden haber producido en el mensaje.

Para facilitar la corrección de los errores tras la transmisión, codificamos el mensaje introduciendo información redundante siguiendo unas reglas sistemáticas. Un ejemplo de codificación es el de triplicar cada secuencia que se envía. Esta información codificada es la que realmente se envía, y gracias a esta redundancia el receptor puede detectar, y en algunos casos corregir, los errores que se hayan podido producir, recuperando así el mensaje original.

2.2. Códigos lineales

Un bloque de código C es un conjunto de M palabras

$$C = \{c_1, c_2, \ldots, c_M\}$$
$$c_i = (c_{i0}, c_{i1}, \ldots, c_{in-1})$$

donde las palabras son n-tuplas. Denominamos n a la longitud del código. Los elementos c_{ij} pertenecen a un alfabeto finito de q símbolos.

Habitualmente nos vamos a encontrar con códigos lineales, que se describen como espacios vectoriales. Supongamos que $\mathbb{F}$ es un cuerpo y n un número natural, entonces los elementos de $\mathbb{F}^n$ se pueden ver como un espacio vectorial $V = (\mathbb{F}^n, +, \mathbb{F})$ donde

$$x, y \in \mathbb{F}^n$$
$$x = (x_0, x_1, \ldots, x_{n-1})$$
$$y = (y_0, y_1, \ldots, y_{n-1})$$
$$x + y = (x_0 + y_0, x_1 + y_1, \ldots, x_{n-1} + y_{n-1})$$
$$\lambda x = (\lambda x_0, \lambda x_1, \ldots, \lambda x_{n-1}) \text{ con } \lambda \in \mathbb{F}$$

Recordemos que un espacio vectorial tiene una base, es decir, un conjunto de vectores linealmente independientes, y que el número de elementos de la base indica la dimensión del espacio vectorial.

Definición 2.1. Un bloque de código lineal C $[n, k]$, es un subespacio k-dimensional del espacio vectorial V.

El vector nulo siempre es una palabra. El número de palabras es $M = q^k$, siendo q el número de elementos del cuerpo $\mathbb{F}$.

Cuando un código contiene información, ésta se encuentra en forma de una larga secuencia de números binarios. La secuencia se divide en bloques de longitud k, cada uno de los cuales denominaremos u. Por lo tanto, necesitamos una función que codifique estos vectores en palabras del código. En la *codificación sistemática*, las primeras k coordenadas de las palabras del código son los elementos de cada bloque u, mientras que las restantes $n - k$ coordenadas de la palabra se denominan símbolos de paridad, nombre que justificaremos más adelante.

2.2.1. Matrices generatriz y de paridad

Un código lineal C que codifica k bits de información en un espacio de código de n bits se representa mediante una matriz generatriz $G_{k\times n}$ cuyas entradas son todas elementos de $\mathbb{F}_q$, más particularmente, ceros y unos. El mensaje de k bits x se codifica como xG. Recordemos que un código que emplea n bits para codificar k bits de información se denomina un código $[n, k]$. El conjunto de palabras del código para éste corresponden al espacio vectorial generado por las columnas de G, por lo que para que todos los mensajes tengan una codificación única, es necesario que las columnas de G sean linealmente independientes.

Definición 2.2. Una matriz generatriz G de un código C $[n, k]$ es una matriz $k \times n$ cuyas filas son linealmente independientes. Es una base del espacio vectorial que forma el código.

El mismo código puede describirse mediante distintas matrices generatrices o bases del espacio vectorial. La matriz G tiene k filas, de manera que, mediante las operaciones entre filas y columnas necesarias, podemos conseguir que k columnas formen una matriz identidad $I_{k\times k}$. Podemos asumir que esta matriz puede elegirse como las primeras k columnas, quedando la matriz generatriz

$$G = (I, A)$$

La corrección de errores para los códigos lineales se entiende más fácilmente introduciendo una formulación alternativa de los códigos lineales mediante la matriz de paridad. Así, un código $[n, k]$ se define como todos los vectores x de n elementos sobre el cuerpo $\mathbb{F}_q$ que cumplen $Hx = 0$, donde hemos definido la matriz de paridad como la matriz $H_{n-k,n}$. En este caso, H debe de cumplir que sus filas sean linealmente independientes.

Definición 2.3. Una matriz de paridad H para un código C $[n, k]$ es una matriz $(n - k) \times n$ cuyas filas son linealmente independientes.

De la misma manera que en la matriz generatriz, podemos realizar operaciones entre filas para escribir la matriz de paridad como

$$H = \left(-A^T, I\right)$$

donde I representa la matriz identidad de dimensiones $(n - k) \times (n - k)$. De esta manera, las últimas $(n - k)$ coordenadas de las palabras del código se obtienen como combinaciones lineales de los primeros k elementos de la matriz de paridad, siendo ésta la causa de que se denominen símbolos de paridad.

La matriz de paridad también recibe el nombre de matriz de control debido a que controla si un elemento está o no en el código, simplemente multiplicándolo por la derecha.

Proposición 2.4. Sea y la palabra recibida. Entonces $y \in C$ si y sólo si $Hy^T = 0$.

Para conectar la imagen de la matriz de paridad para los códigos lineales con la imagen de la matriz generatriz, necesitamos desarrollar un proceso que nos permita pasar de una a otra.

- Para pasar de la matriz de paridad a la matiz generatriz, elegimos k vectores linealmente independientes $y_1, \ldots, y_k$ generando el núcleo de H, y hacemos que G tenga como columnas de y_1 a y_k.

- Para pasar de la matriz generatriz a la matriz de paridad, tomamos $n-k$ vectores linealmente independientes $y_1, \ldots, y_{n-k}$ ortogonales a las columnas de G, y hacemos que las filas de H sean $y_1^T, \ldots, y_{n-k}^T$.

La matriz de paridad facilita la detección de errores y la recuperación del mensaje original. Supongamos que codificamos el mensaje x como $y = xG$, pero se produce un error e debido al ruido que corrompe y, generando la palabra del código corrupta $y' = y + e$. Dado que por la proposición 2.4 sabemos que $Hy = 0$ para todas las palabras del código, obtenemos que $Hy' = He$. Hy' se denomina síndrome del error, y contiene información sobre el error que se ha producido, permitiendo recuperar la palabra original y.

Las definiciones de las matrices generatriz y de paridad de un código nos permiten explicar una construcción importante de códigos, conocida cono la construcción dual. Supongamos que C es un código $[n, k]$ con matriz generatriz G y matriz de paridad H. Podemos definir otro código, el dual de C, que denotamos como $C^\perp$, que tiene como matriz generatriz H^T y como matriz de paridad G^T. Es decir, el dual de C se compone de todas las palabras y que cumplen que son ortogonales a todas las palabras de C. Un código se denomina débilmente autodual si $C \subseteq C^\perp$, y estrictamente autodual si $C = C^\perp$. La construcción dual para códigos lineales clásicos se introduce en el estudio de la corrección de errores cuántica, y es la clave para la construcción de una importante clase de códigos lineales, conocida como códigos CSS.

2.2.2. Distancia mínima y peso mínimo

Definición 2.5. El peso de Hamming de un vector x, que denotaremos como $w_H(x)$, es igual al número de coordenadas no nulas en x.

En ocasiones el peso de Hamming se denomina simplemente peso. Para una palabra recibida $y' = y + e$, el número de errores que hayan ocurrido es igual al peso del vector e.

Definición 2.6. El peso mínimo de un código, w, es el peso de Hamming mínimo entre todas las palabras del código.

Definición 2.7. La distancia de Hamming entre dos vectores x e y, que denotaremos como $d_H(x, y)$, es el número de coordenadas en las que difieren.

Como en el caso del peso, la distancia de Hamming puede denominarse simplemente distancia. Podemos comprobar que d_H satisface las propiedades usuales de una distancia

$$\begin{aligned} &d_H(x,y) = 0 \Leftrightarrow x = y \\ &d_H(x,y) = d_H(y,x) \\ &d_H(x,y) \leq d_H(x,z) + d_H(z,y) \end{aligned}$$

Con esta definición de la distancia, V pasa a ser un espacio métrico.

Definición 2.8. La distancia de un código es la distancia de Hamming mínima entre dos palabras del código cualesquiera

$$d(C) = \min_{x,y \in C, x \neq y} d(x,y)$$

Lema 2.9. Dado un código lineal, $x + y$ es una palabra del código si y sólo si x e y lo son, de manera que obtenemos

$$d(C) = \min_{x \in C, x \neq 0} wt(x)$$

Definiendo $d \equiv d(C)$, decimos que C es un código $[n, k, d]$. La importancia de la distancia es que un código con distancia como mínimo $2t + 1$ para algún entero t, es capaz de corregir errores hasta en t bits, decodificando el mensaje corrupto y' como la única palabra del código y que satisface $d(y, y') \leq t$.

Lema 2.10. En un código $[n, k]$ la distancia mínima es igual al peso mínimo de una palabra no nula.

Algunos códigos cortos pueden tener distancias mínimas sencillas de calcular, como los siguientes ejemplos, que son particularmente importantes.

Definición 2.11. Un código de Hamming binario es un código cuya matriz de paridad tiene como columnas m vectores binarios no nulos.

La longitud de un código de Hamming binario es $2^m - 1$ y la dimensión $2^m - 1 - m$. La distancia mínima es siempre 3.

Definición 2.12. Un código de Hamming binario extendido se obtiene añadiendo una columna nula a la matriz de paridad de un código de Hamming, y a continuación añadiendo una fila de todo unos.

Por lo tanto, la longitud del código extendido es 2^m. Todas las palabras del código extendido tienen peso par, y la distancia mínima es 4. Es decir, los parámetros $[n, k, d]$ del código de Hamming binario extendido son $[2^m, 2^m - m - 1, 4]$.

Podemos entender cómo conseguir realizar la corrección de errores en códigos lineales empleando el concepto de distancia. Supongamos que x e y son palabras de n bits cada una. Como ya hemos visto, la distancia de Hamming $d(x, y)$ entre x e y se define como el número

de posiciones en las que x e y difieren, mientras que el peso de Hamming de una palabra x se define como el número de posiciones con valor no nulo de x. Por lo que se obtiene que $d(x,y) = wt(x+y)$. Veamos la conexión con la corrección de errores, suponiendo que codificamos x como $y = xG$ empleando un código lineal de corrección de errores. El ruido corrompe los bits codificados, dando lugar a la palabra $y' = y + e$. Dado que la probabilidad de que se produzca un error es menor de $1/2$, la palabra que es más probable que haya sido codificada es la palabra y que minimiza el número de errores necesarios para pasar de y a y', es decir, la que minimiza $wt(e) = d(y,y')$. En la práctica, este método es ineficiente, puesto que para determinar la distancia mínima $d(y,y')$ suele ser necesario recorrer las 2^k palabras posibles y.

Nos interesa conocer además algunas condiciones que nos permitan saber si existen o no códigos con parámetros particulares. Un conjunto de estas condiciones es conocido como la cota de Gilbert-Varshamov, que sostiene que para valores grandes de n, existe un código corrector de errores $[n,k]$ que protege contra los errores que se producen en t bits para algún k, de manera que

$$\frac{k}{n} \geq 1 - H\left(\frac{2t}{n}\right)$$

donde

$$H(x) \equiv -x\log(x) - (1-x)\log(1-x)$$

es la entropía binaria de Shannon. La importancia de esta cota es que garantiza la existencia de códigos buenos, siempre que no se intenten codificar demasiados bits k en un número muy pequeño de bits n.

2.3. Códigos Reed-Solomon

En primer lugar, vamos a ver una cota superior en la distancia mínima de cualquier código.

Teorema 2.13. La cota de Singleton
Sea C un código $[n,k]$ con distancia mínima d. Entonces

$$d \leq n - k + 1$$

Definición 2.14. Códigos Reed-Solomon
Sean $x_1, \ldots, x_n$ distintos elementos de un cuerpo finito $\mathbb{F}_q$. Para $k \leq n$ consideremos el conjunto $\mathbb{P}_k$ de los polinomios en $\mathbb{F}_q[x]$ de grado menor que k. Un código Reed-Solomon está formado por las palabras del código

$$(f(x_1), f(x_2), \ldots, f(x_n))$$

donde $f \in \mathbb{P}_k$.

Podemos observar que la longitud del código es $n \leq q$. El código es lineal, puesto que

$$\left.\begin{array}{rcl} c_1 & = & (f_1(x_1), \ldots, f_1(x_n)) \\ c_2 & = & (f_2(x_1), \ldots, f_2(x_n)) \end{array}\right\} \quad \Rightarrow \quad ac_1 + bc_2 = (g(x_1), \ldots, g(x_n))$$

donde $a, b \in \mathbb{F}_q$ y $g(x) = af_1(x) + bf_2(x)$.

Teorema 2.15. Sean $a(x), b(x) \in \mathbb{F}[x],\ b \neq 0$. Entonces existen los polinomios únicos $q(x)$ y $r(x)$ con $\deg(r(x)) < \deg(b(x))$ tales que

$$a(x) = q(x)\, b(x) + r(x)$$

Los polinomios en $\mathbb{P}_k$ forman un espacio vectorial sobre $\mathbb{F}_q$ de dimensión k, puesto que existen k coeficientes. Aplicamos ahora un teorema fundamental del álgebra (teorema 2.15): dos polinomios distintos no pueden generar la misma palabra del código, puesto que la diferencia sería un polinomio de grado menor que k y no puede tener n ceros, por lo que la dimensión del código es k. El peso de una palabra será como mínimo $n - k + 1$ puesto que el polinomio de grado menor que k puede tener como máximo $k - 1$ ceros. Combinando esto con el teorema 2.13 obtenemos

Teorema 2.16. La distancia mínima de un código Reed-Solomon $[n, k]$ es $n - k + 1$.

En multitud de aplicaciones se toma $x_i = \alpha^{i-1},\ i = 1, 2, \ldots, q-1$, donde α es un elemento primitivo de $\mathbb{F}_q$, por lo que en este caso tenemos $x_i^n = 1,\ i = 1, \ldots, n$ y $n = q - 1$.

Teorema 2.17. Un código Reed-Solomon C evalúa polinomios de grado menor que k y tiene parámetros $[n, k, n - k + 1]$, mientras que su dual $C^\perp$ evalúa polinomios de grado menor que $n - k$ y tiene parámetros $[n, n - k, k + 1]$.

2.4. Códigos cíclicos

Los códigos cíclicos son los más utilizados en la corrección de errores. En esta sección denotaremos los vectores como $x = (x_0, x_1, \ldots, x_{n-1})$.

2.4.1. Noción de código cíclico

Definición 2.18. Un código lineal $C\ [n, k]$ es cíclico si para cada $(c_0, c_1, \ldots, c_{n-1}) \in C$ se verifica $(c_{n-1}, c_0, \ldots, c_{n-2}) \in C$.

Es decir, exigimos que C sea invariante por permutaciones cíclicas.

Corolario 2.19. Dado un código cíclico $C\ [n, k]$, existe un único polinomio mónico $g(X) \in \mathbb{F}[X]$ divisor de $X^n - 1$, tal que $C = \langle g(X) \rangle$ es el ideal generado por $g(X)$.

2.4.2. Matrices generatriz y de paridad

Proposición 2.20. Sea C un código cíclico $[n,k]$ con polinomio generador $g(X)$ de grado $n-k$, el conjunto

$$\left\{g(X), Xg(X), \ldots, X^{k-1}g(X)\right\}$$

es una base de C.

Corolario 2.21. Un código cíclico C $[n,k]$ con polinomio generador $g(X) = g_0 + g_1X + \cdots + g_{n-k}X^{n-k}$ tiene matriz generatriz

$$G = \begin{pmatrix} g_0 & g_1 & g_2 & \cdots & g_{n-k} & & & & & \\ & g_0 & g_1 & \cdots & g_{n-k-1} & g_{n-k} & & & & \\ & & \ddots & \ddots & & \ddots & \ddots & & & \\ & & & & \ddots & \ddots & & \ddots & \ddots & \\ & & & & & g_0 & g_1 & \cdots & g_{n-k-1} & g_{n-k} \end{pmatrix}$$

Definición 2.22. Si C es un código cíclico $[n,k]$, con polinomio generador $g(X)$ de grado $n-k$, el polinomio de paridad de C será

$$h(X) = \frac{X^n - 1}{g(X)} = h_0 + h_1X + \cdots + h_kX^k$$

Proposición 2.23. Siguiendo con la definición anterior, podemos obtener la matriz de control de C

$$H = \begin{pmatrix} & & & & h_k & h_{k-1} & \cdots & h_1 & h_0 \\ & & & h_k & h_{k-1} & h_{k-2} & \cdots & h_0 & \\ & & \iddots & \iddots & & \iddots & \iddots & & \\ & \iddots & \iddots & & \iddots & \iddots & & & \\ h_k & h_{k-1} & \cdots & h_1 & h_0 & & & & \end{pmatrix}$$

Se puede observar que $h(X)$ es un generador de $C^{\perp}$, lo que demuestra que el dual de un código cíclico también es cíclico.

Una amplia clase de los códigos Reed-Solomon son códigos cíclicos.

Capítulo 3

Fundamentos de la Computación Cuántica

La computación cuántica y la información cuántica es el estudio del trabajo de procesado de la información que puede ser realizado por sistemas mecánicos cuánticos. La Mecánica Cuántica es un conjunto de reglas empleadas para la construcción de teorías físicas. Estas reglas son simples, pero van en contra de nuestra intuición, por lo que tanto la computación como la información cuántica tienen su origen en los esfuerzos de comprender mejor la Mecánica Cuántica. El objetivo de este estudio es desarrollar herramientas que nos permitan mejorar nuestra intuición sobre ese campo.

Por ejemplo, a principio de 1980 se pensaba que se podían emplear los efectos cuánticos para enviar información a una velocidad mayor que la de la luz, lo cual contradecía la Teoría de la Relatividad de Einstein. Esto podría producirse si se pudieran clonar los estados cuánticos. Sin embargo, como veremos más adelante, ésto no es posible, por lo que se sigue cumpliendo la Teoría de la Relatividad.

Un objetivo que ha contribuido al desarrollo de la computación y la información cuánticas es el interés en obtener un control completo sobre sistemas cuánticos simples, es decir, con un único elemento. Desde 1970 se han desarrollado muchas técnicas para controlar sistemas cuánticos sencillos, como puede ser el caso de atrapar un único átomo y aislarlo del resto del mundo, pudiendo así probar muchos aspectos distintos de su comportamiento con una gran precisión.

La manera de relacionar la Mecánica Cuántica con la computación e información cuánticas es la Ciencia de la Computación, que se ha desarrollado a lo largo de miles de años, puesto que sus orígenes se extienden hasta los tiempos de Hammurabi (1750 a.c.), cuando se empezaron a desarrollar algunas ideas de algoritmos muy sofisticados. Conforme han avanzado los tiempos, se han conseguido crear máquinas que sean capaces de implementar diversos algoritmos, llegando hasta los ordenadores actuales.

Sin embargo, el desarrollo de la computación empieza a encontrar problemas para seguir minimizando el tamaño de los aparatos electrónicos, debido a que empiezan a interferir algunos efectos cuánticos en su funcionamiento. Una posible solución a este problema sería cambiar el paradigma de la computación actual. Un paradigma alternativo sería el que propone emplear la Mecánica Cuántica en la computación en sustitución de la Física Clásica. De esta manera se podrían realizar simulaciones de manera eficiente que no son posibles de implementar en la computación clásica eficientemente. Cuando decimos que un algoritmo es eficiente nos referimos a que se ejecuta en un tiempo polinomial del tamaño del problema resuelto. Por otra parte, cuando un algoritmo es ineficiente necesita un tiempo que normalmente es exponencial.

Para entender cómo funciona la computación cuántica, es necesario comprender en primer lugar algunos fundamentos de la Mecánica Cuántica, por lo que vamos a dedicar este capítulo a explicar estos aspectos. [7][10][11][13]

3.1. Bits cuánticos

En computación clásica, el concepto fundamental es el bit. Análogamente, en computación cuántica se emplea el objeto matemático bit cuántico o qubit. Los bits clásicos se encuentran en un estado 0 ó 1, y los qubits se comportan de manera similar.

3.1.1. Sistemas con un único qubit

Dos posibles estados de los qubits son el $|0\rangle$ y el $|1\rangle$. Sin embargo, los qubits pueden encontrarse en otros estados, siendo estos superposición de los dos anteriores

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

Donde α y β son números complejos, aunque en muchas ocasiones pueden considerarse números reales. Por lo tanto, podemos considerar que el estado de un qubit es un vector en un espacio vectorial complejo bidimensional. De esta manera, los estados $|0\rangle$ y $|1\rangle$ se pueden entender como una base de estados computacionales, formando así una base ortonormal de este espacio vectorial.

Al intentar conocer el estado de un qubit nos encontramos con el siguiente problema: al realizar una medida, no podemos determinar el estado cuántico de un qubit, sino que éste se decantará por uno de los dos estados de la base. Es decir, si medimos el qubit $|\psi\rangle$ que hemos definido anteriormente, obtendremos como resultado el estado $|0\rangle$ con probabilidad $|\alpha|^2$, o el estado $|1\rangle$ con probabilidad $|\beta|^2$. Por supuesto, $|\alpha|^2 + |\beta|^2 = 1$, dado que la suma de las probabilidades debe ser la unidad. Por lo tanto, podemos considerar que el estado del qubit es un vector unitario en el espacio vectorial complejo bidimensional, puesto que su norma debe ser igual a 1.

Una vez vista la superposición en la que se puede encontrar el estado de un qubit, podemos llegar a la conclusión de que los qubits pueden existir en un continuo de estados entre $|0\rangle$ y $|1\rangle$. Un estado particular que vamos a emplear en varias ocasiones es el siguiente

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

Al medir este qubit obtendremos como restultado el estado $|0\rangle$ la mitad de las veces y el $|1\rangle$ la otra mitad. Este estado se denota como $|+\rangle$, mientras que el estado $|-\rangle$ es el que vemos a continuación

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

Existen varios experimentos físicos mediante los que se consigue llevar a los electrones del estado $|0\rangle$ al $|1\rangle$, como puede ser el caso de excitar sus niveles energéticos mediante fotones, donde se conocen la energía y tiempo necesarios para llevar a los electrones a un estado intermedio, el $|+\rangle$.

Otro problema que encontramos al determinar el estado de un qubit es que, al realizar la medida, el qubit pasa a encontrarse en el estado medido. Por lo tanto, la medida cambia el estado del qubit, y lo colapsa, pasando de encontrarse en una superposición de estados de la base a ser uno de estos estados de la base. Este comportamiento es uno de los postulados fundamentales de la Mecánica Cuántica.

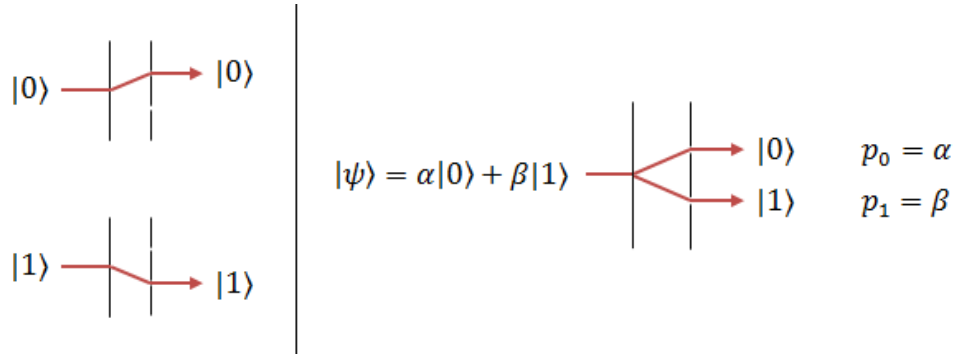

Figura 1: *Izquierda*: Medida del estado de un bit clásico. *Derecha*: Medida del estado de un bit cuántico.

Por lo tanto, aunque los qubits pueden contener infinita información, puesto que hay infinitas superposiciones posibles, sólo podemos obtener bits individuales de información sobre su estado. Sólo si pudiéramos acceder a una fuente inagotable de qubits idénticos podríamos conocer los valores de α y β.

También es interesante mencionar que la medida del estado de un qubit se puede realizar en otras bases distintas a la computacional. Por ejemplo, podemos tomar como base los estados $|+\rangle$ y $|-\rangle$, de manera que podemos reescribir el estado $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ como

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

Y al realizar la medida podríamos obtener como resultado el estado $|+\rangle$ con probabilidad $|\alpha+\beta|/2$ o $|-\rangle$ con probabilidad $|\alpha-\beta|/2$.

En general, dada una base de estados $|a\rangle$ y $|b\rangle$, podemos expresar un estado arbitrario como una combinación lineal de esta base $\alpha|a\rangle + \beta|b\rangle$. Si los estados son ortonormales, podemos realizar una medida con respecto a la base $|a\rangle$ y $|b\rangle$, obteniendo como resultado $|a\rangle$ con probabilidad $|\alpha|^2$ o $|b\rangle$ con probabilidad $|b|^2$.

3.1.2. Sistemas con varios qubits

Supongamos que tenemos dos qubits. Si se tratara de dos bits clásicos, los posibles estados serían

$$00 \qquad 01 \qquad 10 \qquad 11$$

Por lo tanto, en el caso de los bits cuánticos lo que encontramos es una sistema con cuatro elementos en la base de estados computacionales

$$|00\rangle \qquad |01\rangle \qquad |10\rangle \qquad |11\rangle$$

Un par de qubits puede existir en una superposición de estos cuatro estados, empleando una vez más coeficientes complejos, que en muchas ocasiones podremos asumir reales

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

En este caso, el resultado de medir el sistema de qubits será x, donde $x = 00, 01, 10, 11$, con probabilidad α_x. Una vez realizada la medida, el estado del sistema pasará a ser el medido, $|x\rangle$. En este caso, la condición de normalización es la siguiente

$$\sum_{x\in\{0,1\}^2} |\alpha_x|^2 = 1$$

Sin embargo, en este caso realizar la medida no es tan sencillo. Para ello, debemos medir cada qubit por separado. Supongamos que medimos en primer lugar, el primer qubit, y obtenemos el estado $|0\rangle$ con probabilidad $|\alpha_{00}|^2 + |\alpha_{01}|^2$. El estado del qubit tras la medida será

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

La función del denominador es normalizar este nuevo estado, satisfaciendo así la condición de normalización existente.

Un estado importante en un sistema de dos qubits es el estado de Bell o par EPR

$$\frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right)$$

Su importancia radica en que la medida del segundo qubit siempre coincide con la del primero, es decir, la medida está correlacionada.

3.2. Puertas lógicas

Los cambios que suceden en los estados de los qubits se pueden describir mediante la Computación Cuántica. Los ordenadores cuánticos se construyen mediante circuitos cuánticos que consisten en un conjunto de cables y puertas lógicas cuánticas, permitiéndonos manipular la información cuántica.

3.2.1. Puertas lógicas en sistemas con un único qubit

Algunas puertas lógicas clásicas tienen su análogo en la computación cuántica (véase referencia [11]). Es el caso de la puerta NOT. Existen varios procesos que llevan a los qubits del estado $|0\rangle$ al $|1\rangle$ y viceversa. Sin embargo, saber que ésto se produce en la base de los estados cuánticos no es suficiente para conocer qué ocurre en el caso de tener una superposición de estados sin conocer más profundamente las propiedades de las puertas lógicas cuánticas.

De hecho, la puerta cuántica NOT actúa de manera lineal, llevando el estado

$$\alpha|0\rangle + \beta|1\rangle$$

al estado en el que el papel de los estados $|0\rangle$ y $|1\rangle$ se han intercambiado

$$\alpha|1\rangle + \beta|0\rangle$$

Este comportamiento lineal es una propiedad general de la Mecánica Cuántica, puesto que si no se diera podríamos encontrar paradojas como la posibilidad de viajar en el tiempo, comunicaciones a mayor velocidad que la luz, y la violación de la segunda ley de la Termodinámica.

Podemos representar la puerta lógica cuántica NOT en forma de matriz

$$X \equiv \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

Escribiendo el estado cuántico de un qubit mediante notación vectorial

$$\begin{bmatrix} \alpha \\ \beta \end{bmatrix}$$

de manera que la primera fila corresponde al estado $|0\rangle$ y la segunda al $|1\rangle$, entonces al aplicar la puerta NOT el estado resultante es

$$X \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \beta \\ \alpha \end{bmatrix}$$

Por lo tanto, las puertas lógicas en sistemas con un único qubit pueden escribirse en forma de matrices cuadradas 2×2. La condición de normalización se debe cumplir también tras la actuación de una puerta lógica, de manera que en el estado $|\psi'\rangle = \alpha'|0\rangle + \beta'|0\rangle$ debe mantener la propiedad $|\alpha'|^2 + |\beta'|^2 = 1$. Ésto nos indica otra propiedad de estas matrices, y es que han de ser unitarias, es decir, $U^\dagger U = I$, donde $U^\dagger$ es el adjunto de U e I es la matriz identidad 2×2.

La unitariedad es la única condición que deben cumplir las puertas cuánticas. De ahí que existan muchas puertas no triviales de un único qubit, en contraste con las puertas clásicas donde la única puerta no trivial de un único bit es el NOT. Dos puertas importantes que se emplean un gran número de veces son la puerta Z

$$Z \equiv \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

que no cambia el estado $|0\rangle$ y cambia el signo del estado $|1\rangle$, y la puerta de Hadamard

$$H \equiv \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

Esta puerta cambia el estado $|0\rangle$ al estado $|+\rangle$ que habíamos presentado anteriormente, y el estado $|1\rangle$ al $|-\rangle$. Es interesante observar que H^2 no es una puerta, puesto que $H^2 = I$, de manera que si aplicamos dos veces H a un estado, éste se mantiene sin cambios.

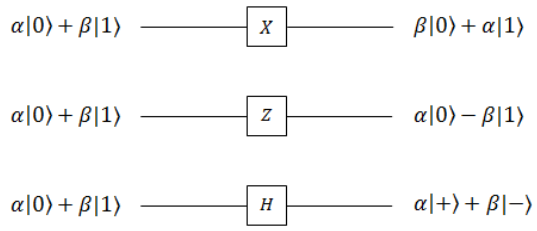

Figura 2: Puertas lógicas para sistemas con un único qubit.

3.2.2. Puertas lógicas en sistemas con varios qubits

El prototipo de puerta lógica cuántica para múltiples qubits es la puerta $CNOT$. Esta puerta lógica tiene dos elementos de entrada, el qubit de control (representado en la línea superior) y el qubit objetivo (en la línea inferior).

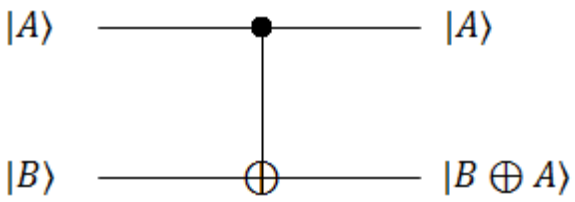

Figura 3: Puerta lógica $CNOT$ para sistemas con varios qubits.

Esta puerta funciona de la siguiente manera: si el qubit de control se encuentra en el estado $|0\rangle$, el objetivo se queda como está. Sin embargo, si el de control se encuentra en el $|1\rangle$, el objetivo cambia de estado.

$$|00\rangle \rightarrow |00\rangle \qquad |01\rangle \rightarrow |01\rangle \qquad |10\rangle \rightarrow |11\rangle \qquad |11\rangle \rightarrow |10\rangle$$

También podemos verlo como una generalización de la puerta XOR clásica, o en forma de matriz unitaria

$$U_{CN} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Una propiedad de las puertas cuánticas es que siempre son reversibles, por lo que una puerta puede ser siempre invertida por otra. Además, se ha demostrado que cualquier puerta lógica para un sistema con múltiples qubits puede descomponerse en una puerta $CNOT$ y una puerta lógica para sistemas con un único qubit.

3.3. Circuitos cuánticos

Vamos a estudiar más detalladamente los componentes de los circuitos cuánticos. Un ejemplo de circuito simple con tres puertas lógicas es el que observamos en la imagen 4.

Los circuitos se leen de izquierda a derecha. Cada línea representa un cable del circuito cuántico. Sin embargo, estos cables no tienen por qué ser físicos, pueden representar el paso del tiempo, o de una partícula como un fotón moviéndose de un lugar a otro del espacio, por ejemplo.

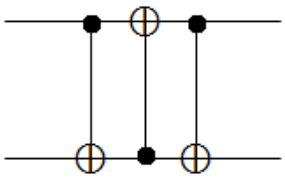

Figura 4: Circuito para intercambiar dos qubits.

El circuito representado se emplea para intercambiar los estados de los dos qubits. Veamos cómo actúa este circuito sobre dos qubits con sus estados en una base de estados computacional $|a, b\rangle$.

$$\begin{aligned} |a, b\rangle &\rightarrow |a, b \oplus a\rangle \\ &\rightarrow |a \oplus (b \oplus a), b \oplus a\rangle = |b, b \oplus a\rangle \\ &\rightarrow |b, (b \oplus a) \oplus b\rangle = |b, a\rangle \end{aligned}$$

Existen algunos aspectos de los circuitos clásicos que no se aceptan en los cuánticos. Por ejemplo, no está permitido realizar "loops", es decir, volver desde una parte posterior del circuito a otra anterior, por lo que decimos que los circuitos son acíclicos. Además, no se permite que se unan dos cables simultáneamente a un tercero, puesto que esta operación no sería reversible. Por otra parte, tampoco está permitido el caso contrario, que de un cable salgan dos, puesto que se producirían copias del qubit y la Mecánica Cuántica lo prohíbe.

Podemos necesitar introducir nuevas puertas lógicas, por lo que vamos a ver otra convención sobre los circuitos cuánticos. Supongamos que U es cualquier matriz unitaria que actúa sobre n qubits, de manera que podemos entender que U es una puerta cuántica para estos qubits. Podemos definir una puerta CU como extensión de la ya conocida $CNOT$. Esta puerta tendría un único qubit de control y n qubits objetivo, de manera que si el qubit de control se encuentra en el estado $|0\rangle$ no sucede nada y si se encuentra en el $|1\rangle$ se aplica la puerta U a los objetivos.

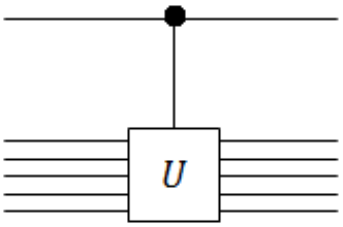

Figura 5: Puerta CU.

Un ejemplo típico de puerta CU es la puerta $CNOT$, que es una puerta CU con $U = X$.

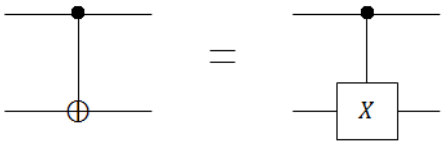

Figura 6: Dos representaciones distintas para la puerta $CNOT$.

Otra operación importante es la medida, que se representa mediante un símbolo de medidor, como se observa en la imagen 7. Esta operación convierte un qubit con estado $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ en un bit clásico probabilístico que denominaremos M, que se distingue de los qubits mediante el dibujo de un cable con una línea doble, y toma el valor 0 con probabilidad $|\alpha|^2$ o 1 con probabilidad $|\beta|^2$.

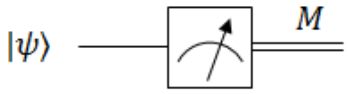

Figura 7: Símbolo en los circuitos cuánticos para la medida.

3.3.1. ¿Es posible copiar qubits?

La puerta $CNOT$ nos va a permitir responder a esta pregunta, demostrando esta propiedad fundamental de la información cuántica.

En primer lugar estudiemos cómo se copian los bits clásicos. Para ello se emplea una puerta $CNOT$ clásica. Se toma como bit de control aquel que se quiere copiar y como bit objetivo uno inicializado a 0, como se observa en la imagen 8. Así obtenemos como resultado dos bits idénticos.

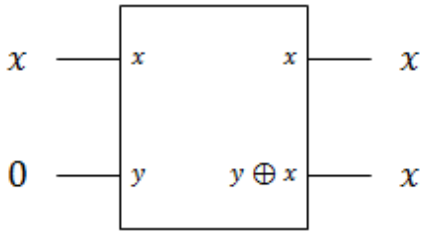

Figura 8: Circuito clásico para copiar un bit desconocido.

Supongamos que intentamos copiar un qubit con estado desconocido $|\psi\rangle = a|0\rangle + b|1\rangle$ empleando la puerta $CNOT$ cuántica. Al unir los dos qubits el estado que obtenemos es el siguiente

$$[a|0\rangle + b|1\rangle]\,|0\rangle = a|00\rangle + b|10\rangle$$

Recordemos que la puerta $CNOT$ niega el segundo qubit cuando el primero se encuentra en el estado $|1\rangle$, de manera que el estado final del qubit objetivo es $a|00\rangle + b|11\rangle$. Por lo tanto, si lo que intentamos copiar es un qubit con estado $|\psi\rangle = |0\rangle$ o $|\psi\rangle = |1\rangle$, es decir, si se comporta como un bit clásico, obtenemos una copia del estado, lo que significa que los circuitos cuánticos pueden copiar información clásica codificada como $|0\rangle$ o $|1\rangle$. Sin embargo, para un estado general $|\psi\rangle$ sabemos que

$$|\psi\rangle|\psi\rangle = a^2|00\rangle + ab|01\rangle + ab|10\rangle + b^2|11\rangle$$

Por lo tanto, a menos que $ab = 0$, el circuito de la imagen 9 no copia el estado inicial.

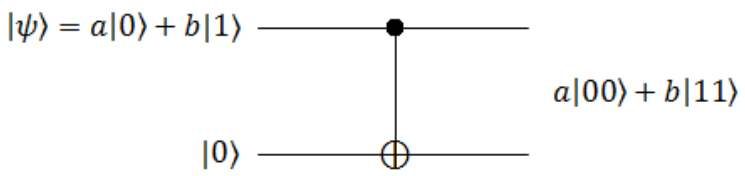

Figura 9: Circuito cuántico para "copiar" un qubit desconocido.

De hecho, es imposible crear una copia de un estado cuántico desconocido. Esta propiedad de la imposibilidad de copiar qubits se conoce como el teorema de la no-clonación, y es una de las diferencias principales entre la información clásica y la cuántica.

3.3.2. Ejemplo: Teleportación cuántica

La teleportación cuántica es una técnica que permite mover estados de un lugar a otro, incluso si no existe ningún canal cuántico que comunique el lugar desde donde se envía con el lugar de recepción. Como curiosidad, en julio de 2017 un grupo de científicos chinos consiguieron enviar un paquete de información desde el Tibet hasta un satélite en órbita que se encontraba a 1400 kilómetros de la superficie terrestre, batiendo así el récord de la distancia más larga recorrida mediante la teleportación.

Veamos cómo funciona. Supongamos que Alice y Bob generan un par EPR del que cada uno coge un qubit y se separan. Un tiempo después, Alice necesita mandar un qubit $|\psi\rangle$ a Bob, sin conocer su estado y pudiendo enviar únicamente información clásica.

Si estudiamos la situación de Alice, lo que intenta hacer es muy complicado, prácticamente imposible. Por un lado, sólo dispone de un qubit $|\psi\rangle$, por lo que no puede realizar medidas ni otros procesos para determinar su estado. Además, aunque conociera el estado, al encontrarse los qubits en un continuo entre los estados $|0\rangle$ y $|1\rangle$, necesitaría una cantidad infinita de información clásica para poder describirlo.

Sin embargo, gracias a la teleportación cuántica, empleando el par EPR que comparten puede enviar a Bob el qubit $|\psi\rangle$ sin necesitar una gran cantidad de información clásica.

En rasgos generales, los pasos que deben seguir son los siguientes: Alice debe hacer que interactuen el qubit $|\psi\rangle$ con su qubit del par EPR y entonces medir los dos qubits, de manera que puede obtener cuatro posibles resultados clásicos: 00, 01, 10 y 11. Una vez realizada la medida, envía a Bob el resultado que haya obtenido. Dependiendo de la información clásica que Bob haya recibido de Alice, realiza una de las cuatro posibles operaciones en su par EPR (estas operaciones las veremos más adelante). Así, puede obtener el estado $|\psi\rangle$ original. El circuito presentado en la imagen 10 nos da una descripción del circuito cuántico necesario para implementar esta teleportación cuántica. Las dos líneas superiores representan el sistema de Alice, mientras que la línea inferior es el sistema de Bob.

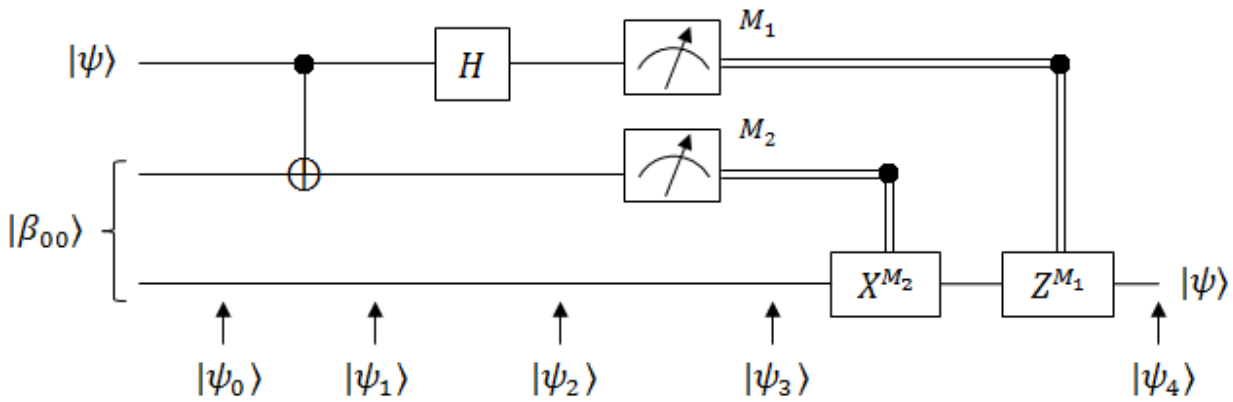

Figura 10: Circuito cuántico para la teleportación de un qubit.

Vamos a ver de manera más precisa qué ocurre en este circuito. El estado $|\beta_{00}\rangle$ es uno de los denominados estados de Bell que tiene como valor

$$|\beta_{00} = \frac{|00\rangle + |11\rangle}{\sqrt{2}}$$

.

El estado que queremos teleportar es el $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, donde α y β son desconocidos. Por lo tanto, el estado de entrada al circuito es

$$|\psi_0\rangle = |\psi\rangle|\beta_{00}\rangle = \frac{1}{\sqrt{2}}\left[\alpha|0\rangle\left(|00\rangle + |11\rangle\right) + \beta|1\rangle\left(|00\rangle + |11\rangle\right)\right]$$

donde vamos a emplear la convención de que los dos primeros qubits corresponden al sistema de Alice y el tecero al de Bob. A continuación, Alice envía sus qubits a través de una puerta $CNOT$

$$|\psi_1\rangle = \frac{1}{\sqrt{2}}\left[\alpha|0\rangle\left(|00\rangle + |11\rangle\right) + \beta|1\rangle\left(|10\rangle + |01\rangle\right)\right]$$

Tras esto, Alice envía el primer qubit a través de una puerta de Hadamard

$$|\psi_2\rangle = \frac{1}{2}\left[\alpha\left(|0\rangle + |1\rangle\right)\left(|00\rangle + |11\rangle\right) + \beta\left(|0\rangle - |1\rangle\right)\left(|01\rangle + |01\rangle\right)\right]$$

Reagrupando términos, el estado en este punto del circuito queda

$$|\psi_2\rangle = \frac{1}{2}\left[|00\rangle\left(\alpha|0\rangle + \beta|1\rangle\right) + |01\rangle\left(\alpha|1\rangle + \beta|0\rangle\right) + |10\rangle\left(\alpha|0\rangle - \beta|1\rangle\right) + |11\rangle\left(\alpha|1\rangle - \beta|0\rangle\right)\right]$$

Así escrita, la expresión se divide en cuatro términos dependiendo de los qubits que tenga Alice. Al realizar la medida, Alice puede obtener cualquiera de los cuatro conjuntos de la expresión, 00, 01, 10 o 11, determinando el estado de Bob tras la medida:

$$\begin{array}{rcl} 00 & \mapsto & |\psi_3\left(00\right)\rangle \equiv \left[\alpha|0\rangle + \beta|1\rangle\right] \\ 01 & \mapsto & |\psi_3\left(01\right)\rangle \equiv \left[\alpha|1\rangle + \beta|0\rangle\right] \\ 10 & \mapsto & |\psi_3\left(10\right)\rangle \equiv \left[\alpha|0\rangle - \beta|1\rangle\right] \\ 11 & \mapsto & |\psi_3\left(11\right)\rangle \equiv \left[\alpha|1\rangle - \beta|0\rangle\right] \end{array}$$

Es decir, dependiendo de la medida de Alice, el qubit de Bob termina en uno de estos cuatro estados posibles. Para saber en cuál se encuentra, Bob necesita que Alice se lo comunique. Hasta aquí, la teleportación podría producirse a una velocidad superior a la de la luz, lo que según la Teoría de la Relatividad, implicaría que la información podría viajar hacia atrás en el tiempo. Sin embargo, Alice necesita un canal de comunicación clásico para enviar el resultado de su medida a Bob, y estos canales están limitados por la velocidad de la luz, de manera que la teleportación cuántica no puede alcanzar velocidades mayores.

Para finalizar, una vez que Bob conoce la medida obtenida por Alice, puede arreglar su estado aplicando la puerta lógica necesaria para recuperar el estado $|\psi\rangle$. Si el estado medido por Alice es el 00, Bob no necesita hacer nada; si es el 01, debe aplicar la puerta X; para el 10, la puerta a aplicar es la Z; y por último, si es el 11, debe aplicar en primer lugar la puerta X y después la Z.

Un último punto a observar es que con la teleportación cuántica parece violarse el teorema de la no-clonación. Sin embargo, esto no es así, debido a que al finalizar el proceso el único qubit que se encuentra en el estado $|\psi\rangle$ es el qubit objetivo.

Capítulo 4

Códigos Cuánticos

En este capítulo vamos a estudiar cómo el ruido puede modificar la información cuántica introduciendo errores, y desarrollaremos la base de la teoría de códigos cuánticos correctores de errores, protegiendo así la información cuántica del ruido y los consecuentes errores. Como ejemplo, tanto los módems como los lectores de CD emplean códigos de corrección de errores para protegerse de los efectos del ruido.

En este desarrollo de la corrección cuántica de errores vamos a asumir que tanto la codificación como la decodificación de la información de los estados cuánticos de los qubits se realiza sin errores en todo momento. Esto puede suceder en la realidad si para la codificación y decodificación se emplean ordenadores cuánticos aislados casi completamente del ruido, mientras que será imposible de conseguir si las puertas cuánticas empleadas en la codificación y decodificación presentan ruido propio.[1][7][8][9][12][14][16][17][18][19]

4.1. Tipos de errores y cómo corregirlos

La idea principal es proteger el mensaje que se quiere enviar de los efectos del ruido. Para ello, al codificar el mensaje debemos introducir información redundante, de manera que aunque alguna parte del mensaje sufra cambios por el efecto del ruido en el canal de comunicación, al estar la información repetida a lo largo de la codificación, se pueda recuperar el mensaje original completo al decodificarlo.

Veamos esta idea de manera sencilla empleando los bits clásicos. Supongamos que queremos enviar un bit de un lugar a otro empleando un canal de comunicación clásico con ruido. El efecto del ruido será cambiar el valor del bit con probabilidad p ($0 < p < 1$), mientras que el bit mantendrá su valor con probabilidad $1 - p$. Este tipo de canales se denomina canal binario simétrico, y se representa como se puede observar en la imagen 11.

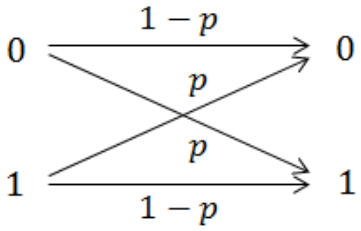

Figura 11: Canal binario simétrico.

Una manera sencilla de proteger al bit del efecto del ruido es sustituirlo por tres copias suyas

$$\begin{aligned} 0 &\rightarrow 000 \\ 1 &\rightarrow 111 \end{aligned}$$

Las cadenas de bit que sustituyen al original, 000 y 111, se suelen denominar 0 lógico y 1 lógico. De esta manera, el receptor obtiene tres bits y debe decidir qué valor es el del bit original. Suponiendo que la probabilidad de que suceda un cambio en el valor del bit sea $p < 0{,}5$, si los bits tienen valores distintos, el valor del bit original será el que tengan los dos que sean iguales. Veámoslo con un ejemplo supongamos que recibimos a la salida del canal la cadena 010. Como la probabilidad de que los bits cambien no es muy alta, lo más probable es que el bit que haya cambiado sea el segundo, de manera que el bit enviado sería el 0.

Este tipo de decodificación se denomina voto mayoritario, puesto que se elige como valor final el que presentan el mayor número de bits a la salida. El voto mayoritario es correcto si sólo se ha cambiado un bit, mientras que falla si los bits cambiados son dos o más. La probabilidad de que dos o más bits cambien es $3p^2\,(1-p)+p^2$, por lo que la probabilidad de error de éste método es $p_e = 3p^2 - 2p^3$. Sin esta codificación, la probabilidad de error es p, de manera que siempre que $p < 0{,}5$ se cumple que $p_e < p$. Este tipo de código se denomina código de repetición, y su idea principal es que el código tenga la suficiente información redundante como para poder reconstruir el código inicial una vez se han producido los errores debidos al ruido.

Para proteger los estados cuánticos del ruido, queremos desarrollar códigos cuánticos de corrección de errores basándonos en principios similares al caso clásico. Sin embargo, encontramos algunas diferencias entre estos dos tipos de códigos, siendo tres las principales dificultades a las que tenemos que hacer frente. En primer lugar, los qubits no pueden clonarse, por lo que no podemos implementar un código cuántico de repetición como hacemos para los bits clásicos. Además, aunque se diera el caso de que pudiéramos clonarlos, no es posible medir y comparar los tres estados cuánticos que se obtienen a la salida del canal. Por otra parte, los errores que se pueden producir son continuos, y determinar qué error ha sucedido para corregirlo requeriría una precisión infinita, junto con unos recursos también infinitos. Por último, al realizar la medida de los qubits, destruimos la información cuántica que contienen, de manera que la reconstrucción del estado inicial se convierte en algo imposible. Por suerte, vamos a demostrar que estos problemas se pueden solventar.

4.1.1. Código de inversión del bit

Supongamos que enviamos qubits a través de un canal que mantiene sus estados originales con probabilidad $1-p$, mientras que cambia su estado con probabilidad p. Es decir, existe una probabilidad p de que el qubit con estado $|\psi\rangle$ pase a encontrarse en el estado $X|\psi\rangle$, donde X es el operador σ_x de Pauli, al que vamos a denominar operador de inversión del bit. A este canal se le denomina canal de inversión del bit, y vamos a explicar el código de inversión de bit, capaz de proteger a los qubits de los efectos del ruido en estos canales.

Supongamos que codificamos un qubit con estado $a|0\rangle + b|1\rangle$ es tres qubits como $a|000\rangle + b|111\rangle$. Es decir, la codificación se escribiría como

$$\begin{aligned} |0\rangle &\rightarrow |0_L\rangle \equiv |000\rangle \\ |1\rangle &\rightarrow |1_L\rangle \equiv |111\rangle \end{aligned}$$

La notación $|0_L\rangle$ y $|1_L\rangle$ indica que se trata del $|0\rangle$ lógico y del $|1\rangle$ lógico, no de los estados físicos $|0\rangle$ y $|1\rangle$. Un circuito para implementar esta codificación se puede ver en la parte izquierda de la imagen 12, antes de que se envíe a través del canal de inversión de bit E_{bit}.

Supongamos que hemos conseguido codificar perfectamente el estado $a|0\rangle + b|1\rangle$ como $a|000\rangle + b|111\rangle$. Cada uno de los qubits se envía a través de copias independientes del canal de inversión de bit. Supongamos ahora que se produce una inversión de bit en uno o ninguno de los qubits. Existe un proceso de corrección de errores simple con únicamente dos pasos que nos permite recuperar el estado original del qubit. Estudiemos cómo funciona este proceso:

1. *Detección del error o diagnóstico de síndrome.* Realizamos una medida que nos permite conocer qué error ha ocurrido en el estado cuántico, si es que se ha producido alguno. El resultado que se obtiene de esta medida se denomina el síndrome de error. Para el canal de inversión de bit existen cuatro síndromes de error, que se corresponden con cuatro operadores

$$\begin{aligned} P_0 &\equiv |000\rangle\langle 000| + |111\rangle\langle 111| && \text{sin inversión} \\ P_1 &\equiv |100\rangle\langle 100| + |011\rangle\langle 011| && \text{inversión de bit en el primer qubit} \\ P_2 &\equiv |010\rangle\langle 010| + |101\rangle\langle 101| && \text{inversión de bit en el segundo qubit} \\ P_3 &\equiv |001\rangle\langle 001| + |110\rangle\langle 110| && \text{inversión de bit en el tercer qubit} \end{aligned}$$

   Supongamos por ejemplo que se produce una inversión de bit en el segundo qubit, de manera que el estado tras la alteración es $a|010\rangle + b|101\rangle$. En este caso

$$\begin{aligned} \langle\psi|P_0|\psi\rangle &= 0 & \langle\psi|P_2|\psi\rangle &= 1 \\ \langle\psi|P_1|\psi\rangle &= 0 & \langle\psi|P_3|\psi\rangle &= 0 \end{aligned}$$

   de manera que el síndrome de error es 2. Es importante observar que la medida del síndrome no cambia el estado de los qubits, de manera que el síndrome únicamente contiene información sobre qué error se ha producido.

2. *Recuperación.* Empleamos el valor del síndrome de error para saber qué proceso hemos de seguir para recuperar el estado inicial. Por ejemplo, hemos obtenido que el síndrome de error es 2, lo que nos indica que se ha producido una inversión del segundo bit, de manera que invirtiendo el qubit de nuevo podemos recuperar el estado original $a|000\rangle + b|111\rangle$. Los cuatro posibles síndromes y el proceso que ha de seguirse en cada caso son los siguientes:

   - Síndrome 0, es decir, no se ha producido ningún error. En este caso, no es necesario hacer nada, ya hemos obtenido el estado original.
   - Síndrome 1, se ha producido una inversión del bit en el primer qubit. Debemos invertir de nuevo el primer qubit.
   - Síndrome 2, la inversión del bit ha ocurrido en el segundo qubit. Invirtiéndolo recuperamos el estado inicial.
   - Síndrome 3, encontramos una inversión del bit en el tercer qubit. Volvemos a invertirlo para completar el proceso.

   De esta manera, para cualquier error del síndrome recuperamos el estado original de manera completamente precisa.

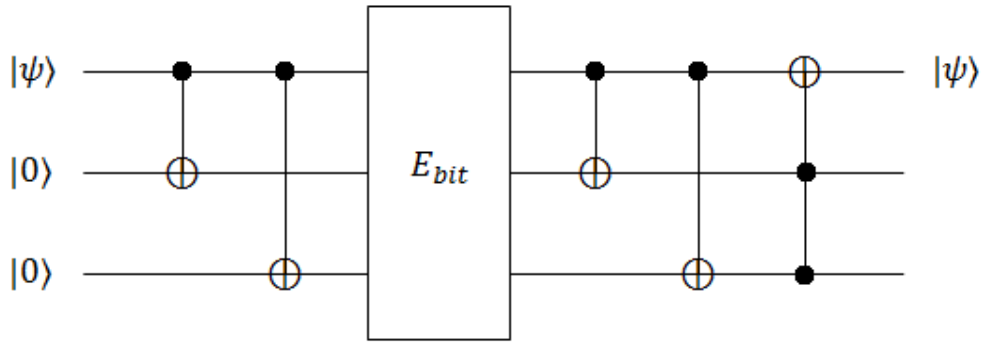

Figura 12: Circuito cuántico del código de inversión del bit.

Este procedimiento funciona de manera perfecta si los errores se producen en uno o ningún qubit. Ésto ocurre con una probabilidad $(1-p)^3 + 3p\,(1-p)^2 = 1 - 3p^2 + 2p^3$. La probabilidad de que no se corrija un error es de $3p^2 - 2p^3$, la misma que habíamos obtenido en el caso del código de repetición clásico. De la misma manera, dado $p < 0{,}5$, la codificación y decodificación mejoran la fiabilidad del almacenamiento del estado cuántico.

Existe otra manera de entender la medida del síndrome útil para generalizar el código de tres qubits. Supongamos que en lugar de emplear las cuatro proyecciones P_0, P_1, P_2 y P_3, realizamos dos medidas, la primera con el observable Z_1Z_2 (es decir, $Z \otimes Z \otimes I$), y la segunda con Z_2Z_3. Cada uno de estos observables tienen autovalores ± 1, de manera que cada medida produce un bit de información. De la medida completa obtenemos dos bits, es decir, cuatro posibles síndormes de error. La primera medida se puede entender como la comparación entre los dos primeros qubits y la segunda como la comparación entre los dos últimos. La medida de Z_1Z_2 da como resultado $+1$ si los qubits son iguales, o -1 si son distintos. De la misma manera, la medida de Z_2Z_3 obtiene como resultado $+1$ si el segundo y tercer qubits son iguales, o -1 en el caso contrario. Combinando las dos medidas podemos saber si se ha producido una inversión del bit o no y dónde:

- Si ambos resultados son $+1$, entonces no se ha producido ninguna inversión de bit.
- Si la medida de Z_1Z_2 da $+1$ y la de Z_2Z_3 da -1, el tercer qubit ha sido invertido.
- Si el resultado de medir Z_1Z_2 es -1 y el de medir Z_2Z_3 es $+1$, lo más probable es que únicamente el primer qubit haya sido invertido.
- Si el resultado de ambas medidas es -1, el invertido habrá sido el segundo qubit.

Es importante recalcar que ninguno de los dos procesos de medida da información sobre los valores de a y b ni destruyen la superposición de estados cuánticos.

4.1.2. Código de inversión del signo

Otro canal cuántico ruidoso interesante es el que da lugar al modelo de inversión de signo para sistemas de un único qubit. En este modelo, el qubit se mantiene en su estado original con probabilidad $1 - p$, mientras que con probabilidad p la fase relativa de los estados $|0\rangle$ y $|1\rangle$ es invertida. Esto se traduce en que el operador de inversión de signo Z se aplica al qubit con probabilidad $p > 0$, de manera que el estado $a|0\rangle + b|1\rangle$ pasa a ser $a|0\rangle - b|1\rangle$ bajo la inversión del signo.

Es sencillo convertir un canal de inversión del signo en uno de inversión del bit. Supongamos que trabajamos con la base de qubits

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

$$|-\rangle = \tfrac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right)$$

En esta base, el operador Z convierte el estado $|+\rangle$ en el estado $|-\rangle$ y viceversa, es decir, actúa como un operador de inversión del bit en la base de estados $|+\rangle$ y $|-\rangle$.

Por lo tanto, podemos emplear los estados $|0_L\rangle \equiv |++\rangle$ y $|1_L\rangle \equiv |--\rangle$ como los estados cero y uno lógicos. Las operaciones necesarias para proteger al código de los errores de inversión del signo que puede inducir el ruido se implementan del mismo modo que las de los canales de inversión del bit, pero con respecto a la base $|+\rangle$, $|-\rangle$ en lugar de hacerlo respecto de $|0\rangle$, $|1\rangle$. Para ello empleamos las puertas de Hadamard y sus inversas (que también son puertas de Hadamard) en los puntos estratégicos que se observan en la imagen 13, puesto que la puerta de Hadamard es la que nos permite pasar los estados de la base $|0\rangle$, $|1\rangle$ a la base $|+\rangle$, $|-\rangle$.

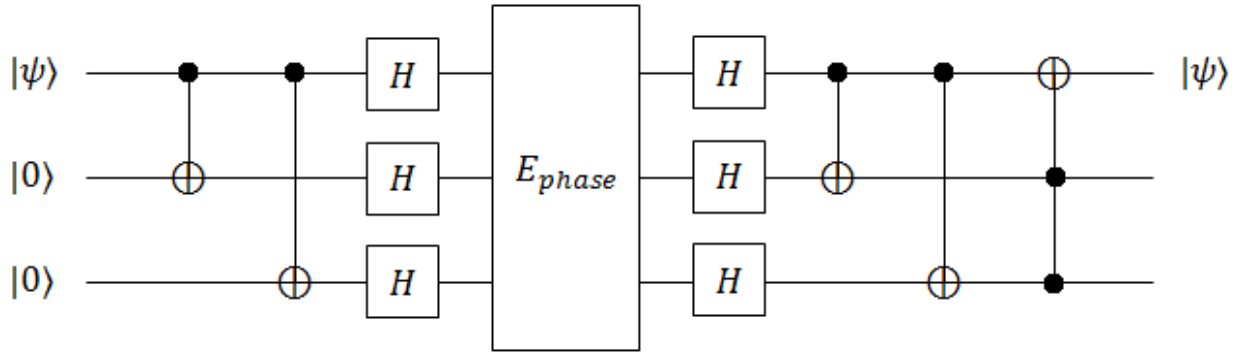

Figura 13: Circuito cuántico del código de inversión del signo.

La codificación de los canales de inversión del signo se implementa según los siguientes dos pasos:

1. Codificamos en tres qubits, de la misma manera que en los canales de inversión del bit.
2. Aplicamos una puerta de Hadamard a cada qubit.

La detección de los errores se puede realizar aplicando las medidas mediante los proyectores de la misma manera que en el canal de inversión del bit, pero conjugándolos con las puertas de Hadamard:

$$P_j \;\rightarrow\; P_j' \equiv H^{\otimes 3} P_j H^{\otimes 3}$$

De manera equivalente, la medida del síndrome se puede realizar midiendo los observables

$$\begin{aligned} H^{\otimes 3} Z_1 Z_2 H^{\otimes 3} &= X_1 X_2 \\ H^{\otimes 3} Z_2 Z_3 H^{\otimes 3} &= X_2 X_3 \end{aligned}$$

Es interesante interpretar estas medidas en líneas similares a la medida de Z_1Z_2 y Z_2Z_3 para los códigos de inversión del bit. Medir los observables X_1X_2 y X_2X_3 corresponde a comparar el signo de los dos primeros qubits y el de los dos últimos.

Por último, la corrección de errores se completa con la operación de recuperación, que es la operación de recuperación de los códigos de inversión del bit conjugada con una puerta de Hadamard.

Este código para los canales de inversión del signo tiene las mismas características que el código para los canales de inversión del bit. Estos dos canales son unitariamente equivalentes, puesto que existe un operador unitario U, que en este caso es la puerta de Hadamard, tal que la acción de un canal es la misma que la del otro si el primer canal va precedido por U y seguido por $U^\dagger$.

4.2. El código de Shor

El código de Shor permite proteger códigos cuánticos simples de los efectos producidos por un error arbitrario en un qubit. Este código es una combinación de los códigos de inversión del bit y de inversión del signo para tres qubits. En primer lugar, codificamos el qubit empleando el código de inversión del signo

$$\begin{aligned} |0\rangle &\rightarrow |++\rangle \\ |1\rangle &\rightarrow |--\rangle \end{aligned}$$

A continuación se codifica cada qubit mediante el código de inversión del bit para tres qubits

$$|+\rangle \rightarrow \tfrac{1}{\sqrt{2}}\left(|000\rangle + |111\rangle\right)$$

$$|-\rangle \rightarrow \tfrac{1}{\sqrt{2}}\left(|000\rangle - |111\rangle\right)$$

De esta manera, obtenemos como resultado un código de nueve qubits, cuyas palabras vienen dadas por

$$|0\rangle \rightarrow |0_L\rangle \equiv \tfrac{1}{2\sqrt{2}}\left(\left(|000\rangle + |111\rangle\right)\left(|000\rangle + |111\rangle\right)\left(|000\rangle + |111\rangle\right)\right)$$

$$|1\rangle \rightarrow |1_L\rangle \equiv \tfrac{1}{2\sqrt{2}}\left(\left(|000\rangle - |111\rangle\right)\left(|000\rangle - |111\rangle\right)\left(|000\rangle - |111\rangle\right)\right)$$

El circuito cuántico que codifica y decodifica el código de Shor se muestra en la imagen 14. Podemos observar que la primera parte de la codificación corresponde a la codificación del código de inversión del signo para tres qubits. Tras esto, cada qubit se codifica empleando el código de inversión del bit. Este método de codificar empleando una estructura de niveles se conoce como concatenación.

Gracias a esta codificación, el código de Shor es capaz de proteger cualquier bit de los errores de inversión del bit y del signo. Además, supongamos que se producen ambos errores en el mismo qubit. En ese caso, se puede observar que el proceso de detección de un error de inversión del bit detectaría un error y lo corregiría, y a continuación el proceso de detección de un error de inversión del signo lo detectaría también y lo corregiría. Por lo tanto, el código de Shor también permite corregir errores combinados de inversión del bit y del signo en un mismo qubit.

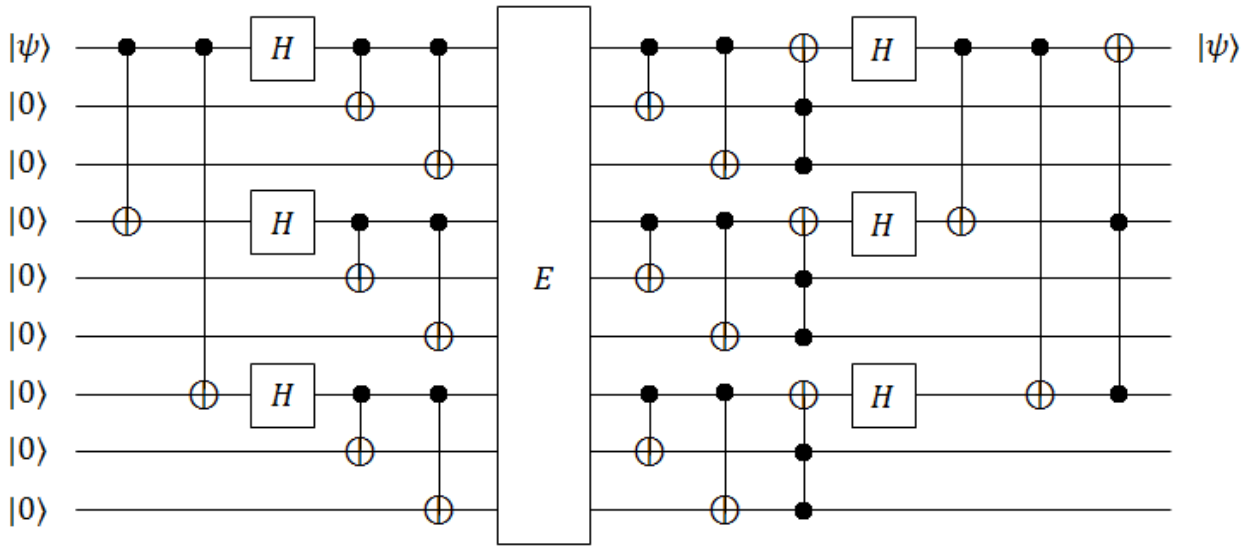

Figura 14: Circuito cuántico del código de Shor.

Además, se puede comprobar que este código protege también de errores completamente arbitrarios, suponiendo que éstos afecten a un único qubit. Podemos observar que no necesitamos un trabajo adicional para proteger contra estos errores arbitrarios, puesto que el proceso actual es suficiente. Éste es un ejemplo del hecho de que lo que parece un continuo de errores que pueden ocurrir a un único qubit se puede corregir si conseguimos corregir un subconjunto discreto de estos errores. Esta discretización de errores es la base del por qué la corrección de errores cuántica funciona. En los códigos clásicos esta discretización no es posible.

4.3. Teoría de corrección de errores cuántica

Las ideas básicas de la teoría de corrección de errores cuántica generalizan de manera natural las ideas introducidas por el código de Shor. Los estados cuánticos son codificados por una operación unitaria en un código de corrección de errores cuántico, que se define formalmente como un subespacio C de un espacio de Hilbert. Es conveniente contar con una notación para el proyector en el espacio de código C, por lo que vamos a emplear la notación P. Para el código de inversión del bit con tres qubits, el proyector sería $P = |000\rangle\langle 000| + |111\rangle\langle 111|$. Tras la codificación, el código es expuesto al ruido, a lo que le sigue la medida del síndrome, que nos permite conocer qué tipo de error se ha producido, es decir, el síndrome de error. Una vez se ha determinado, se lleva a cabo la operación de recuperación, devolviendo el sistema cuántico a su estado original.

Para desarrollar una teoría general de corrección de errores cuántica necesitamos hacer las menores asunciones posibles sobre la naturaleza del ruido y el proceso empleado para realizar la corrección del error. Es decir, no vamos a asumir necesariamente que la corrección del error se produce a partir de un método de dos pasos, detección y recuperación. Tampoco vamos a hacer asunciones sobre el ruido que se produce en los sistemas de qubits o si éste es débil o no. Sólo vamos a emplear dos asunciones generales: el ruido se puede describir mediante una operación cuántica $\mathcal{E}$, y el proceso completo de la corrección de errores se realiza mediante una operación cuántica que preserva la traza a la que denominaremos $\mathcal{R}$ y que recibe el nombre de operación de corrección de errores. Así unimos en una única pieza los dos pasos de detección y recuperación. Para que se realice de manera satisfactoria la corrección de errores, para cualquier estado ρ que se encuentre en el código C se debe cumplir

$$(\mathcal{R} \circ \mathcal{E})(\rho) \propto \rho$$

Las condiciones para la corrección de errores cuántica son un conjunto de ecuaciones que se pueden emplear para ver si un código de corrección de errores cuántico protege de un tipo particular de ruido $\mathcal{E}$.

Teorema 4.1. Condiciones para la corrección de errores cuántica
Sea C un código cuántico y P el proyector en C. Supongamos que $\mathcal{E}$ es una operación cuántica con elementos $\{E_i\}$. Una condición necesaria y suficiente para la existencia de una operación de corrección de errores $\mathcal{R}$ que corrija $\mathcal{E}$ en C es

$$PE_i^\dagger E_j P = \alpha_{ij} P$$

para alguna matriz hermítica α de números complejos.

Denominamos a los elementos $\{E_i\}$ del ruido $\mathcal{E}$ errores, y si existe tal $\mathcal{R}$, diremos que $\{E_i\}$ constituye un conjunto de errores corregible. La verificación de las condiciones para la corrección de errores cuántica es directa, pero es una tarea que precisa tiempo.

4.3.1. Discretización de los errores

Normalmente no sabemos exactamente qué error ha afectado a nuestro sistema cuántico, por lo que sería muy útil que un código específico C y una operación de corrección de errores $\mathcal{R}$ se pudieran emplear para proteger contra una clase completa de procesos ruidosos. Las condiciones para la corrección de errores cuántica se pueden adaptar de manera sencilla para obtener exactamente este tipo de protección.

Teorema 4.2. Supongamos que C es un código cuántico y $\mathcal{R}$ es la operación de corrección de errores que recupera el código tras sufrir un proceso ruidoso $\mathcal{E}$ con elementos $\{E_i\}$. Supongamos que $\mathcal{F}$ es una operación cuántica con elementos $\{F_j\}$ que son combinaciones lineales de los E_i, es decir

$$F_j = \sum_i m_{ji} E_i$$

para alguna matriz m_{ij} de números complejos. Entonces, la operación de corrección de errores $\mathcal{R}$ también corrige los efectos del proceso ruidoso $\mathcal{F}$ en el código C.

Esto nos permite introducir un lenguaje más potente para describir los códigos de corrección de errores cuánticos. En lugar de hablar sobre la clase de procesos ruidosos $\mathcal{E}$ que puede ser corregida por un código C y una operación de corrección de errores $\mathcal{R}$, podemos hablar de un conjunto de errores $\{E_i\}$ que son corregibles. Las condiciones para la corrección de errores cuántica se mantienen para estos operadores

$$PE_iE_i^\dagger P = \alpha_{ij}P$$

Los teoremas 4.1 y 4.2 implican que cualquier proceso ruidoso $\mathcal{E}$ cuyos elementos sean combinaciones lineales de los errores $\{E_i\}$ puede ser corregido por la operación de recuperación $\mathcal{R}$.

Por lo tanto, hemos visto que es posible discretizar los errores cuánticos, es decir, para corregir el continuo de errores posibles en un único qubit es suficiente saber cómo corregir un conjunto finito de errores.

4.3.2. Códigos degenerados

Existe una clase interesante en los códigos cuánticos que presenta una propiedad que no encontramos en los clásicos. Estos códigos se denominan códigos degenerados. La manera más sencilla de ilustrarla es empleando el código de Shor, considerando el efecto de los errores Z_1 y Z_2 en las palabras del código. Como ya hemos visto, el efecto es el mismo en dos palabras distintas, cosa que no puede suceder en los códigos clásicos, donde para dos palabras distintas, dos errores distintos dan resultados diferentes. Este fenómeno de encontrar códigos cuánticos degenerados tiene su lado bueno, pero también su lado malo. Las malas noticias son que algunas de las técnicas demostradas para los códigos clásicos para probar la existencia de cotas en la corrección de errores no pueden aplicarse en el caso de los códigos degenerados. Por otra parte, las buenas noticias son que los códigos cuánticos degenerados son unos de los códigos cuánticos más interesantes, puesto que son capaces de contener más información que los clásicos, dado que distintos errores no deben llevar necesariamente el espacio del código a otro espacio ortogonal. Es posible, aunque aún no se ha demostrado, que esta propiedad permita a los códigos degenerados almacenar la información cuántica de una manera más eficiente que a cualquier código no degenerado.

4.3.3. La cota cuántica de Hamming

Para distintas aplicaciones queremos emplear el "mejor" código cuántico posible. El significado de "mejor" depende de la aplicación. Es por esto que queremos encontrar un criterio para determinar si un código con unas características particulares existe o no. Para ello, vamos a desarrollar la cota cuántica de Hamming, una cota simple que nos permite obtener una visión general de las propiedades de los códigos cuánticos. Esta cota sólo se puede aplicar a códigos no degenerados, pero nos da una idea de cómo deberían ser las cotas más generales.

Supongamos que un código no degenerado se emplea para codificar k qubits en n qubits, de manera que es capaz de corregir errores en cualquier subconjunto de t o menos errores. Supongamos que ocurren j errores, con $j \leq t$. Existen $\binom{n}{j}$ conjuntos de lugares donde pueden producirse errores. En cada uno de estos conjuntos existen tres posibles errores (las tres matrices de Pauli X, Y y Z) que pueden producirse en cada qubit, dando un total de 3^j errores posibles. El número total de errores que pueden producirse en t o menos qubits es

$$\sum_{j=0}^{t} \binom{n}{j} 3^j$$

Para codificar k qubits de una manera no degenerada, cada uno de estos errores debe corresponder a un subespacio ortogonal 2^k-dimensional. Todos estos subespacios deben ajustarse al espacio 2^n-dimensional disponible para n qubits, dando lugar a la siguiente desigualdad

$$\sum_{j=0}^{t} \binom{n}{j} 3^j 2^k \leq 2^n$$

Esta desigualdad se denomina cota cuántica de Hamming.

No todos los códigos cuánticos son no degenerados, por lo que la cota cuántica de Hamming es más útil como una regla general que como una cota rápida para los códigos cuánticos.

4.4. Códigos estabilizadores

Hasta ahora hemos obtenido una estructura teórica para estudiar los códigos de corrección de errores cuánticos, pero no tenemos demasiados ejemplos de este tipo de códigos. Para remediarlo, vamos a explicar cómo podemos emplear las ideas de los códigos lineales clásicos para desarrollare la teoría de los códigos estabilizadores, una clase de códigos más general que los códigos CSS que permite construir una gran variedad de códigos cuánticos.

Los códigos estabilizadores, también conocidos como códigos cuánticos aditivos, son una clase importante de códigos cuánticos cuya construcción es análoga a la de los códigos clásicos lineales. Para entender los códigos estabilizadores es útil desarrollar en primer lugar un formalismo estabilizador que nos permita entender una amplia clase de operaciones en la Mecánica Cuántica. Tras describir el formalismo estabilizador, explicaremos cómo las puertas unitarias y la medida se deben describir empleándolo, y veremos un importante teorema que cuantifica las limitaciones de las operaciones estabilizadoras.

4.4.1. El formalismo estabilizador

Veamos en primer lugar el formalismo estabilizador ilustrado con un ejemplo. Consideremos el estado EPR de dos qubits

$$|\psi\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}}$$

Es fácil comprobar que este estado satisface las siguientes identidades

$$X_1 X_2 |\psi\rangle = |\psi\rangle$$
$$Z_1 Z_2 |\psi\rangle = |\psi\rangle$$

Entonces, se dice que el estado $|\psi\rangle$ es estabilizado por los operadores $X_1 X_2$ y $Z_1 Z_2$. El estado $|\psi\rangle$ es el único estado cuántico que es estabilizado por estos operadores.

La idea básica del formalismo estabilizador es que muchos estados cuánticos pueden describirse de una manera más sencilla trabajando con operadores que los estabilicen, en lugar de trabajar explícitamente con ellos. La clave del poder del formalismo estabilizador se encuentra en el empleo de la teoría de grupos. El grupo de mayor interés es el grupo de Pauli G_n en n qubits. Para un único qubit, el grupo de Pauli se define de la siguiente manera

$$G_1 \equiv \{\pm I, \pm iI, \pm X, \pm iX, \pm Y, \pm iY, \pm Z, \pm iZ\}$$

Este conjunto de matrices forma un grupo bajo la operación de la multiplicación de matrices. Los factores multiplicativos ± 1 y $\pm i$ se incluyen para asegurarse de que el grupo G_1 es cerrado bajo multiplicación, formando así un grupo legítimo.

Ahora podemos definir los estabilizadores de una manera un poco más precisa. Supongamos que S es un subgrupo de G_n, y definamos V_S como el conjunto de n estados de qubits fijos para cada elemento de S. V_S es el espacio vectorial estabilizado por S, y S se denomina estabilizador del espacio V_S, dado que todo elemento de V_S es estable bajo la acción de elementos en S.

Existe un fenómeno general importante, y es que un grupo se puede describir por sus generadores. Un conjunto de elementos $g_1, \ldots, g_l$ en un grupo G se dice que genera el grupo G si todo elemento de G se puede escribir como producto de elementos de la lista $g_1, \ldots, g_l$. Lo denotamos como $G = \langle g_1, \ldots, g_l \rangle$. La ventaja de emplear los generadores para describir grupos es que proporcionan una manera compacta de describir el grupo. Para comprobar que un vector es estabilizado por un grupo S, sólo necesitamos comprobar que es estabilizado por los generadores, puesto que entonces es automáticamente estabilizado por el producto de los generadores, lo que lo convierte en una representación más conveniente.

No todo subgrupo S del grupo de Pauli puede emplearse como el estabilizador de un espacio vectorial no trivial. Por ejemplo, consideremos el subgrupo de G_1 consistente en $\{\pm I, \pm X\}$. Está claro que la única solución a $(-I)\,|\psi\rangle = |\psi\rangle$ es $|\psi\rangle = 0$, por lo que $\{\pm I, \pm X\}$ es el estabilizador del espacio vectorial trivial. Por lo tanto, vamos a estudiar las condiciones necesarias para que S estabilice un espacio vectorial no trivial V_S

1. Los elementos de S deben conmutar.

2. $-I$ no puede ser un elemento de S.

Veremos más adelante que estas condiciones también son suficientes para que V_S sea no trivial.

En la práctica, queremos que los generadores $g_1, \ldots, g_l$ sean independientes, es decir, que si eliminamos algún generador el grupo sea más pequeño

$$\langle g_1, \ldots, g_{i-1}, g_{i+1}, \ldots, g_l \rangle \neq \langle g_1, \ldots, g_l \rangle$$

Existe una manera simple de determinar si un conjunto particular de generadores es independiente, basándonos en la idea de la matriz de comprobación, que se denomina así debido a que juega un papel en la teoría de los códigos estabilizadores similar al de la matriz de paridad en los códigos lineales clásicos.

Supongamos que tenemos el subgrupo $S = \langle g_1, \ldots, g_l \rangle$. Existe una manera muy útil de representar los generadores de S empleando la matriz de comprobación. Se trata de una matriz de dimensiones $l \times 2n$ cuyas filas corresponden a los generadores desde g_1 hasta g_l. La parte izquierda de la matriz contiene unos indicando qué generadores contienen X, y la parte derecha contiene unos indicando qué generadores contienen Z. La presencia de un 1 en ambos lados indica que el generador contienen un Y.

Proposición 4.3. Sea $S = \langle g_1, \ldots, g_l \rangle$ tal que $-I$ no es un elemento de S. Los generadores de g_1 a g_l son independientes si y sólo si las filas de la matriz de comprobación correspondientes son linealmente independientes.

Las siguientes proposiciones nos permiten saber que si S es generado por $l = n - k$ generadores independientes que conmutan, y $-I \notin S$, entonces V_S es de dimensión 2^k.

Proposición 4.4. Sea $S = \langle g_1, \ldots, g_l \rangle$ generado por l generadores independientes y satisfaga $-I \notin S$. Fijado i en el rango $1, \ldots, l$, entonces existe $g \in G_n$ tal que $gg_ig^\dagger = -g_i$ y $gg_jg^\dagger = g_j \ \forall j \neq i$.

Concluimos esta introducción a los elementos más básicos del formalismo estabilizador con la siguiente proposición, que nos permite demostrar que V_S es no trivial si se cumple que S es generado por generadores independientes que conmutan y que $-I \notin S$. Es más, como ya hemos comentado anteriormente, si existen $l = n - k$ generadores, entonces V_S es 2^k-dimensional.

Proposición 4.5. Sea $S = \langle g_1, \ldots, g_{n-k} \rangle$ generado por $n - k$ elementos independientes que conmutan de G_n, y tal que $-I \notin S$. Entonces V_S es un espacio vectorial 2^k-dimensional.

4.4.2. Puertas unitarias y el formalismo estabilizador

El formalismo estabilizador también se puede emplear para describir la dinámica de los espacios vectoriales en un espacio de estados más amplio, bajo una variedad de operaciones cuánticas. Esto es especialmente interesante debido a que nos permite describir códigos cuánticos de corrección de errores empleando el formalismo estabilizador, obteniendo maneras elegantes de entender los efectos del ruido y otros procesos dinámicos en estos códigos. Supongamos que aplicamos una operación unitaria U a un espacio vectorial V_S estabilizado por el grupo S. Sea $|\psi\rangle$ un elemento de V_S. Entonces, para cualquier elemento g de S

$$U|\psi\rangle = Ug|\psi\rangle = UgU^\dagger U|\psi\rangle$$

Por lo tanto, el estado $U|\psi\rangle$ es estabilizado por $UgU^\dagger$, de lo que se deduce que el espacio vectorial UV_S es estabilizado por el grupo $USU^\dagger \equiv \left\{UgU^\dagger \middle| \ g \in S\right\}$. Es más, si $g_1, \ldots, g_l$ generan S, entonces $Ug_1U^\dagger, \ldots, Ug_lU^\dagger$ generan $USU^\dagger$, por lo que para calcular el cambio en el estabilizador únicamente necesitamos comprobar cómo afecta a los generadores del estabilizador.

Para algunas operaciones unitarias U, esta transformación de los generadores toma una forma interesante. Supongamos como ejemplo que aplicamos la puerta de Hadamard a un único qubit. Notemos que

$$HXH^\dagger = Z \qquad HYH^\dagger = -Y \qquad HZH^\dagger = X$$

Podemos deducir que tras aplicar la puerta de Hadamard al estado cuántico estabilizado por Z, el estado resultante será estabilizado por X.

Veamos otro ejemplo: supongamos que tenemos n qubits en un estado cuyo estabilizador es $\langle Z_1, \ldots, Z_n \rangle$. Al aplicar la puerta de Hadamard a cada uno de los n qubits, observamos que el estado final tiene como estabilizador $\langle X_1, \ldots, X_n \rangle$. Lo interesante de este ejemplo es que la descripción usual del estado final requiere 2^n elementos para ser especificado, mientras que con la descripción que nos proporcionan los generadores sólo necesitamos n.

Sin embargo, tras aplicar la puerta de Hadamard sigue sin existir entrelazamiento, por lo que no es sorprendente que podamos obtener una descripción compacta. El formalismo estabilizador presenta muchas más posibilidades, incluida una descripción eficiente de la puerta $CNOT$, que junto con la puerta de Hadamard es capaz de generar entrelazamiento.

Para comprender cómo funciona, consideremos cómo se comportan los operadores X_1, X_2, Z_1 y Z_2 bajo la conjugación con $CNOT$. Denotando como U a la puerta $CNOT$, cuyo qubit de control es el 1 y el objetivo es el 2, obtenemos

$$\begin{aligned} UX_1U^\dagger &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \\ &= X_1X_2 \end{aligned}$$

Cálculos similares demuestran que

$$UX_2U^\dagger = X_2 \qquad UZ_1U^\dagger = Z_1 \qquad UZ_2U^\dagger = Z_1Z_2$$

Para comprobar como se conjuga U con otros operadores en el grupo de Pauli de dos qubits, únicamente necesitamos calcular los productos de los operadores que ya conocemos.

El formalismo estabilizador puede emplearse también para describir una amplia clase de estados entrelazados, incluyendo muchos códigos cuánticos de corrección de errores.

Existen más puertas que se pueden describir mediante el formalismo estabilizador. Una de las más importantes, junto con la puerta de Hadamard y la puerta CNOT que ya hemos estudiado, es la puerta de fase, una puerta para un único qubit cuya definición es la siguiente

$$S = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$$

La acción de la puerta de fase por conjugación con las matrices de Pauli se calcula fácilmente

$$SXS^\dagger = Y \qquad SZS^\dagger = Z$$

Es más, se puede comprobar que cualquier operación unitaria que toma elementos de G_n y los lleva a G_n bajo conjugación se puede componer mediante las puertas de Hadamard, CNOT y de fase. Por definición, decimos que el conjunto de U tal que $UG_nU^\dagger = G_n$ es el normalizador de G_n, y se denota como $N(G_n)$, por lo que el normalizador de G_n es generado por las puertas de Hadamard, CNOT y de fase, por lo que a estas puertas a veces se las denomina puertas normalizadoras.

Teorema 4.6. Supongamos que U es un operador unitario en n qubits con la propiedad de que si $g \in G_n$ entonces $UgU^\dagger \in G_n$. Entonces, excepto por una fase global, U está compuesto por $O(n^2)$ puertas de Hadamard, CNOT y de fase.

Otras puertas que no son las normalizadoras también se pueden describir mediante el formalismo estabilizador, sin embargo es mucho menos conveniente que para los circuitos que sólo contienen puertas normalizadoras, debido a la complejidad que puede llegar a alcanzar la descripción de su acción por conjugación. Tenemos la suerte de que la codificación, decodificación, la detección de errores y la recuperación para códigos cuánticos estabilizadores se puede lograr empleando únicamente las puertas normalizadoras, por lo que el formalismo establizador es muy conveniente para el análisis de este tipo de códigos.

4.4.3. Medida en el formalismo estabilizador

La medida en la base computacional también se puede describir de manera sencilla mediante el formalismo estabilizador. Imaginemos que realizamos una medida de $g \in G_n$, siendo g un operador hermítico. Asumimos, sin pérdida de generalidad, que g es un producto de las matrices de Pauli, sin los factores multiplicativos -1 y $\pm i$. Asumamos también que el sistema se encuentra en un estado $|\psi\rangle$ con estabilizador $\langle g_1, \ldots, g_n\rangle$. El estabilizador del estado puede transformarse de dos maneras posibles bajo la medida:

- g conmuta con todos los generadores del estabilizador.
- g anti-conmuta con uno o más de los generadores del estabilizador. Supongamos que el estabilizador tiene generadores $g_1, \ldots, g_n$, y que g anti-conmuta con g_1. Sin pérdida de generalidad, podemos asumir que g conmuta con $g_2, \ldots, g_n$, puesto que si no conmuta con uno de estos elementos, por ejemplo con g_2, entonces es fácil observar que g conmuta con g_1g_2, y podemos simplemente reemplazar el generador g_2 por g_1g_2 en la lista de generadores del estabilizador.

En el primer caso, se deduce que tanto g como $-g$ son elementos del estabilizador, puesto que como $g_jg|\psi\rangle = gg_j|\psi\rangle = g|\psi\rangle$ para cada generador del estabilizador, entonces $g|\psi\rangle$ se encuentra en V_S, y por tanto es un múltiplo de $|\psi\rangle$. Dado que $g^2 = I$, se deduce que $g|\psi\rangle = \pm|\psi\rangle$, por lo que tanto g como $-g$ deben encontrarse en el estabilizador. Asumamos que g se encuentra en el estabilizador, en el caso de $-g$ la demostración sería análoga. En este caso, $g|\psi\rangle = |\psi\rangle$, por lo que la medida de g es $+1$ con probabilidad uno, por lo que la medida no cambia el estado del sistema, y el estabilizador permanece invariante.

Veamos qué sucede en el segundo caso, cuando g anti-conmuta con g_1 y conmuta con el resto de generadores del estabilizador. Notemos que los autovalores de g son ± 1, por lo que los proyectores de la medida con resultado ± 1 vienen dados por $(I \pm g)/2$, respectivamente. Por tanto, las probabilidades de la medida son las siguientes

$$p(+1) = \text{tr}\left(\frac{I+g}{2}|\psi\rangle\langle\psi|\right)$$

$$p(-1) = \text{tr}\left(\frac{I-g}{2}|\psi\rangle\langle\psi|\right)$$

Sabiendo que $g_1|\psi\rangle = |\psi\rangle$ y que $gg_1 = -g_1 g$, obtenemos que la probabilidad de que la medida obtenga como resultado $+1$ es

$$p(+1) = \text{tr}\left(\frac{I+g}{2}g_1|\psi\rangle\langle\psi|\right) = \text{tr}\left(g_1\frac{I-g}{2}|\psi\rangle\langle\psi|\right)$$

Aplicando la propiedad cíclica de la traza, podemos pasar g_1 al final de la traza y absorberla en $\langle\psi|$, empleando $g_1 = g_1^\dagger$, obteniendo así

$$p(+1) = \text{tr}\left(\frac{I-g}{2}|\psi\rangle\langle\psi|\right) = p(-1)$$

Dado que $p(+1) + p(-1) = 1$, podemos deducir que $p(+1) = p(-1) = 1/2$. Supongamos que el resultado que se obtiene es -1. En ese caso, el nuevo estado del sistema es $|\psi^-\rangle \equiv (I-g)|\psi\rangle/\sqrt{2}$, con estabilizador $\langle -g, g_2, \ldots, g_n\rangle$. Si el resultado obtenido es $+1$, el estabilizador quedaría $\langle g, g_2, \ldots, g_n\rangle$.

4.4.4. El teorema de Gottesman-Knill

Los resultados sobre el empleo de estabilizadores para describir dinámicas unitarias y medidas se puede resumir con el teorema de Gottesman-Knill.

Teorema 4.7. Teorema de Gottesman-Knill
Supongamos que se desarrolla la computación cuántica empleando únicamente los siguientes elementos: preparaciones de estados en la base computacional, puertas de Hadamard, puertas de fase, puertas $CNOT$, puertas de Pauli y medidas de observables en el grupo de Pauli, junto con la posibilidad del control clásico condicionado por los resultados de las medidas. Esta computación se puede simular de manera eficiente en un ordenador clásico.

De manera implícita ya hemos probado este teorema. Los ordenadores clásicos desarrollan la simulación manteniendo un registro de los generadores del estabilizador conforme se van realizando operaciones en la computación. Por ejemplo, para simular la puerta de Hadamard, simplemente actualizamos cada uno de los n generadores describiendo el estado cuántico.

El teorema de Gottesman-Knill subraya la sutilidad del poder de la computación cuántica, mostrando que algunas computaciones cuánticas que implican estados altamente entrelazados se pueden simular de manera eficiente en ordenadores clásicos. Por supuesto, no todas las computaciones cuánticas se pueden describir de manera eficiente mediante el formalismo estabilizador, pero una gran variedad de ellas sí.

4.4.5. Construcción de códigos estabilizadores

La idea básica para describir códigos cuánticos con el formalismo estabilizador es simple: un código estabilizador $[[n, k]]$ se define como el espacio vectorial V_S estabilizado por el subgrupo S de G_n, tal que $-I \notin S$ y estando S generado por $n-k$ generadores independientes que conmutan, $S = \langle g_1, \ldots, g_{n-k} \rangle$. Denotaremos a este código $C(S)$.

Existiendo $n-k$ generadores del estabilizador S, eligiendo cualesquiera 2^k vectores ortonormales del código $C(S)$ obtenemos una base de estados lógica computacional. En la práctica, estos estados se eligen de manera más sistemática, por ejemplo con el método que vemos a continuación. En primer lugar, elegimos los operadores $\bar{Z}_1, \ldots, \bar{Z}_k \in G_n$ tales que $g_1, \ldots, g_{n-k}, \bar{Z}_1, \ldots, \bar{Z}_k$ forman un conjunto independiente que conmuta. El operador $\bar{Z}_j$ hace el papel del operador σ_z lógico de Pauli en el qubit lógico número j, por lo que la base de estados lógica computacional $|x_1, \ldots, x_k\rangle_L$ se define como el estado con estabilizador

$$\langle g_1, \ldots, g_{n-k}, (-1)^{x_1} \bar{Z}_1, \ldots, (-1)^{x_k} \bar{Z}_k \rangle$$

De manera similar definimos $\bar{X}_j$ como el producto de matrices de Pauli que transforman $\bar{Z}_j$ en $-\bar{Z}_j$ bajo conjugación y deja el resto de $\bar{Z}_j$ y g_i invariantes al actuar por conjugación. $\bar{X}_i$ tiene el efecto de una puerta NOT cuántica que actúa sobre el qubit codificado que se encuentra en la posición j. El operador $\bar{X}_j$ satisface $\bar{X}_j g_k \bar{X}_j^\dagger = g_k$, y por lo tanto conmuta con todos los generadores del estabilizador. También es fácil comprobar que $\bar{X}_j$ conmuta con todos los $\bar{Z}_i$ excepto con $\bar{Z}_j$, con el que anti-conmuta.

Vamos a ver cómo se relacionan las propiedades de la corrección de errores de un código estabilizador con los generadores de su estabilizador. Supongamos que codificamos un estado empelando un código estabilizador $C(S)$ $[[n, k]]$ con estabilizador $S = \langle g_1, \ldots, g_{n-k} \rangle$, y se produce un error E que corrompe los datos. Con el siguiente análisis, que consta de tres pasos, vamos a determinar qué tipos de errores se pueden detectar empleando un código $C(S)$ y cuándo se puede recuperar la información. En primer lugar, vamos a ver el efecto de los distintos tipos de error en el espacio del código. Tras esto, estableceremos un teorema general que nos indica qué tipo de errores se pueden detectar y corregir con un código estabilizador. Por último, presentaremos una descripción práctica para realizar la detección del error y su corrección, empleando herramientas que ya conocemos, como es el caso del síndrome de error.

Supongamos que $C(S)$ es un código estabilizador corrompido por un error $E \in G_n$. En el caso de que E anti-conmute con un elemento del estabilizador, E lleva a $C(S)$ a un subespacio ortogonal, y el error en principio puede ser detectado realizando una medida proyectiva apropiada. Si $E \in S$, el error E no corrompe el espacio, por lo que no es demasiado preocupante. El problema lo encontramos cuando E conmuta con todos los elementos de S pero no se encuentra en S, es decir, $Eg = gE \ \forall g \in S$. El conjunto de $E \in G_n$ que cumple la condición anterior se conoce como centralizador de S en G_n, y se denota como $Z(S)$. Para los grupos estabilizadores S que nos interesan, el centralizado es idéntico al normalizador de S, $N(S)$, que consiste en todos los elementos E de G_n tales que $EgE^\dagger \in S \ \forall g \in S$.

Estas observaciones sobre el efecto de los distintos tipos de operadores de error E motivan el establecimiento del siguiente teorema, que traduce las condiciones para la corrección de errores cuántica en términos de los códigos estabilizadores.

Teorema 4.8. Condiciones para la corrección de errores en códigos estabilizadores
Sea S el estabilizador de un código estabilizador $C(S)$. Supongamos que $\{E_j\}$ es un conjunto de operadores en G_n tal que $E_j^\dagger E_k \notin N(S) - S \ \forall j,k$. Entonces $\{E_j\}$ es un conjunto de errores corregibles para el código $C(S)$.

El teorema 4.8 no nos indica de manera explícita cómo podemos llevar a cabo la operación de la corrección de errores. Para entender cómo podemos realizarla, supongamos que $g_1, \ldots, g_{n-k}$ es el conjunto de generadores del estabilizador del código estabilizador $[[n,k]]$, y que $\{E_j\}$ es el conjunto de errores corregibles del código. La detección de errores se realizar midiendo los generadores del estabilizador de g_1 a g_{n-k} en orden, obteniendo así el síndrome del error, que consiste en los resultados de las medidas de β_1 a β_{n-k}. Si el error E_j se produce, entonces el síndrome de error viene dado por β_l, de manera que $E_j g_l E_j^\dagger = \beta_l g_l$. Si E_j es el único operador de error que tiene este síndrome, podemos recuperar el código original aplicando $E_j^\dagger$. Si existen dos errores distintos E_j y $E_{j'}$ que dan lugar al mismo síndrome de error, entonces $E_j P E_j^\dagger = E_{j'} P E_{j'}^\dagger$, donde P es el proyector en el espacio del código, de manera que $E_j^\dagger E_{j'} P E_{j'}^\dagger E_j = P$, por lo que $E_j^\dagger E_{j'} \in S$, y entonces aplicando $E_j^\dagger$ una vez se ha producido el error $E_{j'}$ obtenemos el código original. Por lo tanto, para cada posible síndrome de error, simplemente tenemos que elegir un error E_j con ese síndrome y aplicar $E_j^\dagger$ para conseguir recuperar el código original cuando ese síndrome se observa.

El teorema 4.8 da pie a definir una distancia para los códigos cuánticos análoga a la distancia de los códigos clásicos. Definimos el peso de un error $E \in G_n$ como el número de términos en el producto tensorial que no son iguales a la identidad. La distancia de un código estabilizador $C(S)$ se define como el peso mínimo de un elemento de $N(S) - S$, y si $C(S)$ es un código $[[n,k]]$ con distancia d, entonces decimos que el código $C(S)$ es un código estabilizador $[[n,k,d]]$. Por el teorema 4.8, un código con distancia como mínimo $2t+1$ es capaz de corregir errores arbitrarios en t qubits cualesquiera, del mismo modo que sucedía en el caso clásico.

4.5. Construcción de Códigos Cuánticos

Como continuación de la sección anterior, vamos a centrarnos en dos clases de códigos. En primer lugar, vamos a construir una amplia clase de códigos cuánticos conocida como códigos Calderbank-Shor-Steane (CSS), y a continuación, y para concluir el trabajo, estudiaremos cómo obtener los códigos Reed-Solomon cuánticos (vénase referencias [1] y [12]).

4.5.1. Códigos Calderbank-Shor-Steane

En esta sección vamos a estudiar cómo podemos obtener los códigos Calderbank-Shor-Steane a partir de la modificación y combinación de códigos aditivos. Estos códigos son más conocidos como códigos CSS, y son una importante subclase de la clase más general de códigos estabilizadores.

Teorema 4.9. Supongamos que tenemos dos códigos lineales clásicos C_1 y C_2 con parámetros $[n, k_1, d_1]$ y $[n, k_2, d_2]$ tales que $C_2 \subset C_1$ y C_1 y $C_2^{\perp}$ corrigen t errores. Entonces existe un código cuántico $\mathrm{CSS}(C_1, C_2)$ de parámetros $[[n, k_1 - k_2, d]]$, donde $d \geq \text{mín}\{d_1, d_2\}$, capaz de corregir errores en t qubits, el código CSS de C_1 sobre C_2.

Corolario 4.10. Supongamos que C es un código con parámetros $[n, k, d]$, tal que $C \subset C^{\perp}$. Entonces existe un código cuántico de parámetros $[[n, n - 2k, d']]$, donde $d' \geq d^{\perp}$.

Además, podemos desarrollar códigos cuánticos nuevos a partir de los ya existentes empleando las siguientes construcciones. Sabemos que la suma directa de dos códigos aditivos se define como

$$C \oplus C' = \{uv : u \in C, v \in C'\}$$

Así podemos obtener la suma directa de dos códigos cuánticos de corrección de errores, combinando dos códigos $[[n, k, d]]$ y $[[n', k', d']]$ y obteniendo el código $[[n + n', k + k', d'']]$, donde $d'' = \text{mín}\{d, d'\}$. Un código aditivo que no es una suma directa de otros dos se denomina no-descomponible.

Teorema 4.11. Supongamos que existe un código $[[n, k, d]]$.

1. Si $k > 0$, entonces existe un código $[[n + 1, k, d]]$.
2. Si el código es puro, es decir, si distintos elementos del espacio de errores E producen resultados ortogonales, y $n \geq 2$, entonces existe un código $[[n - 1, k + 1, d - 1]]$.
3. Si $k > 1$ o si $k = 1$ y el código es puro, entonces existe un código $[[n, k - 1, d]]$.
4. Si $n \geq 2$, entonces existe un código $[[n - 1, k, d - 1]]$.
5. Si $n \geq 2$ y el código asociado C contiene un vector de peso 1, entonces existe un código $[[n - 1, k, d]]$.

Si tenemos información adicional sobre C, entonces hay una técnica más potente que en el teorema 4.11 apartado 4 para reducir el código.

Teorema 4.12. Supongamos que tenemos un código lineal $[[n, k, d]]$ con un código asociado $C\ \left(n, 2^{n-k}\right)$. Entonces existe un código lineal $[[n-m, k', d']]$ con $k' \geq k-m$ y $d' \geq d$ para cualquier m tal que exista una palabra de peso m en el dual del código binario generado por el soporte de las palabras de C, donde el soporte de una palabra es el conjunto de posiciones coordenadas en las que la palabra tiene entradas no nulas.

La construcción de la suma directa empleada en el teorema 4.11 apartado 1 se puede generalizar.

Teorema 4.13. Dados dos códigos $[[n_1, k_1, d_1]]$ y $[[n_2, k_2, d_2]]$ con $k_2 \leq n_1$ podemos construir un código $[[n_1 + n_2 - k_2, k_1, d]]$, donde $d \geq \text{mín}\,\{d_1, d_1 + d_2 - k_2\}$.

4.5.2. Códigos Reed-Solomon cuánticos

Como hemos visto en el teorema 2.17, un código Reed-Solomon $[n, k, n-k+1]$ evalúa polinomios de grado menor que k, mientras que su dual $[n, n-k, k+1]$ evalúa polinomios de grado menor que $n-k$.

Para aplicar el corolario 4.10, necesitamos saber cuándo un código Reed-Solomon está contenido en su dual. Para ello, se debe cumplir que $k \leq k-n$.

Teorema 4.14. Sea un código C Reed-Solomon $[n, k, n-k+1]$. Si se cumple que $k \leq n/2$, entonces $C \subset C^{\perp}$. Por lo tanto, existe un código Reed-Solomon cuántico con parámetros $[[n, n-2k, k+1]]$

Bibliografía

[1] A. R. Calderbank, E. M. Rains, P. W. Shor, and N. J. A. Sloane. Quantum Error Correction Via Codes Over GF(4). *IEEE Transactions on Information Theory* 44.4 (1998): 1369-1387.

[2] C. E. Shannon. A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review* 5.1 (2001): 3-55.

[3] R. V. L. Hartley. Transmission of information. *Bell Labs Technical Journal* 7.3 (1928): 535-563.

[4] E. Martínez Moro, C. Munuera Gómez, and D. Ruano Benito. Bases de Gröbner: aplicaciones a la codificación algebraica. Instituto Venezolano de Investigaciones Científicas (2007).

[5] J. Justesen, and T. Høholdt. A Course In Error-Correcting Codes. European Mathematical Society, Zurich, 2004.

[6] R. Lidl, and H. Niederreiter. Introduction to finite fields and their applications. Cambridge University Press, Cambridge, 2nd edition, 1994.

[7] M. A. Nielsen, and I. L. Chuang. Quantum Computation and Quantum Information. Cambridge University Press, Cambridge, 7th edition, 2010.

[8] J. García-López. Códigos cuánticos.

[9] S. J. Devitt, W. J. Munro, and K. Nemoto. Quantum Error Correction for Beginners. *Reports on Progress in Physics* 76.7 (2013): 076001.

[10] W. K. Wootters, and W. Zurek. The no-cloning theorem. *Physics Today* 62.2 (2009): 76-77.

[11] A. Muthukrishnan. Classical and Quantum Logic Gates: An Introduction to Quantum Computing Seminar. (1999).

[12] M. Grassl, W. Geiselmann, and T. Beth. Quantum Reed-Solomon Codes. *AAECC* Vol. 13. No. 1709. 1999.

[13] J. A. Jover. Open quantum systems: geometric description, dynamics and control. Tesis doctoral, Universidad de Zaragoza, Departamento de Física Teórica, 2017.

[14] C. Galindo, O. Geil, F. Hernando, and D. Ruano. On the distance of stabilizer quantum codes given by J-affine variety codes. *Quantum Information Processing* Vol. 16. Issue 4 16:111.

[15] F. Hernando, K. Marshall and M. O'Sullivan. The Dimension of Subcode-Subfields of Shortened Generalized Reed Solomon Codes. *Design Codes and Cryptography* Vol. 69. Issue 1 (2013): 131-142.

[16] C. Galindo, F. Hernando, and D. Ruano. Stabilizer quantum codes from J-affine variety codes and a new Steane-like enlargement. *Quantum Information Processing* Vol. 14. Issue 9 (2015): 3211-3231.

[17] R. Laflamme, et al. Perfect quantum error correcting code. *Physical Review Letters* 77.1 (1996): 198.

[18] M. D. Reed, et al. Realization of three-qubit quantum error correction with superconducting circuits. *Nature 482.7385* (2012): 382-385.

[19] E. Knill, and R. Laflamme. Theory of quantum error-correcting codes. *Physical Review A* 55.2 (1997): 900.