Universidad de Valladolid

Facultad de Ciencias

Trabajo Fin de Grado

Grado en Física

ALGORITMOS CUÁNTICOS CONTROLADOS POR MEDIDA

Autor: Pablo González Tamames

Tutores: Juan Carlos García Escartín y Luis Miguel Nieto

Resumen

En este trabajo se estudian algoritmos cuánticos de búsqueda análogos al algoritmo de Grover pero basados en la medida cuántica. Se empieza con una introducción a física cuántica para luego explicar brevemente en que consiste la computación cuántica. A continuación se explica el algoritmo de Grover desde varios puntos de vista: un pozo de potencial, utilizando el operador inversión sobre la media y como rotaciones en un espacio bidimensional. Para terminar se explica el efecto Zenón cuántico y a partir de los principios de la medida expuestos se plantea una interpretación alternativa del algoritmo de Grover utilizando medidas sucesivas. Finalmente se proponen dos métodos concretos para realizar esas medidas. Ambos sistemas se simulan con Python para comprobar los resultados, comprobando así que los algoritmos basados en medida tienen una eficiencia inferior a los otros puntos de vista dados.

Abstract

In this project we study quantum search algorithms analogous to Grover's algorithm but based on quantum measurement. Firstly, we give an introduction to quantum physics and then briefly explain what quantum computing is about. Then we explain Grover's algorithm from different perspectives: a potential well, using the inversion about average operator and as rotations in a two-dimensional space. Later, the quantum Zeno effect is explained and an alternative interpretation of Grover's algorithm using successive measurements is proposed. Finally, two concrete methods to perform these measurements are proposed. Both systems are simulated with Python to check the results, thus proving that the measurement-based algorithms have a lower efficiency than the other given points of view.

Índice general

Índice de figuras

Capítulo 1

Introducción

En este trabajo se va a tratar de diseñar un algoritmo cuántico cuyo progreso esté dado por sucesivas medidas del sistema. Un algoritmo es una serie de instrucciones que nos ayudan a resolver un problema y el hecho de que sea cuántico significa que utiliza propiedades de la física cuántica para resolver dicho problema. La mecánica cuántica es la rama de la física que explica cómo funciona la naturaleza a pequeña escala. Para comprender este trabajo serán necesarias ciertas nociones sobre física cuántica. En esta introducción se van a presentar de forma breve y más adelante se profundizará más en ellas.

En mecánica cuántica el estado de un sistema viene dado por su función de onda ψ. Esta función de onda viene dada a través del hamiltoniano del sistema y la ecuación de Schrödinger:

$$H\psi = E\psi. \tag{1.1}$$

Esta ψ es una función compleja y calculando su módulo al cuadrado $|\psi|^2$ se puede saber cuál es la probabilidad de encontrar el sistema en una configuración particular. Una propiedad fundamental de los sistemas cuánticos es la posibilidad de crear un estado superposición. Cuando se habla de un estado superposición se está hablando de un sistema que es combinación lineal de varias posibles configuraciones. Sobre este estado en superposición se pueden realizar operaciones pero en el momento en el que dicho sistema se observe quedará completamente determinado en una de las posibles configuraciones que lo formaban. Esta es una propiedad muy útil en computación cuántica, y en particular en este trabajo, ya que cuando hablamos de un algoritmo cuántico controlado por medida nos referimos a observar un estado en superposición y obligar a que este estado quede determinado. Estas propiedades de la mecánica cuántica se tratan más formalmente en el Capítulo 2.

Se ha mencionado ya la expresión "computación cuántica"pero no se ha explicado qué es. Este tipo de computación se distingue de la clásica en el hecho de que utiliza ordenadores cuánticos para operar, que son ordenadores que procesan información utilizando las leyes de la mecánica cuántica. La primera vez que se habló de un ordenador cuántico fue en 1982 y vino de la mano de Richard Feynman. Él introdujo la idea de ordenadores cuánticos porque al ser la naturaleza fundamentalmente cuántica, si se quiere simularla habrá que hacerlo utilizando sus propias leyes.

La diferencia más significativa entre estos ordenadores y los clásicos es la unidad mínima de información. Mientras que en un ordenador clásico esta unidad (bit) solo puede tomar dos valores (0 ó 1), en un ordenador cuántico esta unidad (qubit) es un estado cuántico por lo que puede ser una combinación lineal de ambos. En la sección 2.3 se tratan más las implicaciones que tiene esta diferencia.

En un ordenador clásico es fácil de ver como es la unidad mínima de información: ya que solo puede tomar dos valores será una corriente encendida o apagada. En cuanto a la unidad mínima en computación cuántica se puede representar como un sistema de dos niveles que es algo muy usual en esta rama de la

física. Este sistema puede estar implementado, entre otras maneras, en un átomo en el que se tenga un nivel excitado y el estado fundamental.

En física cuántica es común utilizar una notación especial para simplificar los cálculos, la notación de Dirac. En la sección 2.1 se va dar esta notación con más detalle pero como idea básica comentar que el estado de un partícula o un sistema, su función de onda ψ, en esta notación se representa como $|\psi\rangle$. Al símbolo $|\psi\rangle$ y se le conoce como ket se puede entender como un vector. Por ejemplo los dos estados que forman la combinación lineal que da lugar a la unidad mínima información son $|0\rangle$ y $|1\rangle$:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \qquad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \tag{1.2}$$

También es posible tener un sistema de más de un qubit. En el sistema en este caso viene representado por el producto tensorial de dos estados de un solo qubit:

$$(\alpha\,|0\rangle + \beta\,|1\rangle) \otimes (\delta\,|0\rangle + \gamma\,|1\rangle) = \alpha\delta\,|00\rangle + \alpha\gamma\,|01\rangle + \beta\delta\,|10\rangle + \beta\gamma\,|11\rangle\,, \tag{1.3}$$

donde α, β, δ y γ son números complejos que cumplen la condición de que $|\alpha\delta|^2 + |\alpha\gamma|^2 + |\beta\delta|^2 + |\beta\gamma|^2 = 1$ ya que estos valores representan la probabilidad de, al hacer una medida, obtener cada uno de los estados a los que acompañan y la probabilidad total debe ser 1.

El algoritmo que se trata de replicar en este trabajo es el conocido como algoritmo de Grover, que nos permite encontrar uno o varios elementos de entre un conjunto de N. En este algoritmo se parte de un estado en el que cada una de las N posibilidades está representada en un elemento de un estado del tipo (1.3). Aplicando dos transformaciones a este sistema se consigue que se transfiera probabilidad desde los elementos que no nos interesan hasta los que sí, haciendo así que tras repetir el proceso una serie de pasos, casi con total seguridad se obtenga uno de los elementos que se querían encontrar. En el Capítulo 3 se trata este algoritmo más en profundidad.

En la sección 4.1 se expone un punto de vista del algoritmo que junto con un efecto cuántico expuesto en la sección 4.2 nos lleva a plantear una posible realización del algoritmo con medidas sucesivas de un sistema. En el Capítulo 5 se muestran dos propuestas para una posible implementación del algoritmo junto con los resultados obtenidos gracias a una simulación hecha con Python. Finalmente el trabajo concluye con las conclusiones en el Capítulo 6.

Capítulo 2

Computación clásica frente a computación cuántica

La computación cuántica es un nuevo paradigma de computación que utiliza las propiedades de la mecánica cuántica para resolver problemas. Utilizando este nuevo paradigma se pueden construir una nueva clase de ordenadores (los llamamos ordenadores cuánticos) que, utilizando las propiedades de la mecánica cuántica, son capaces de resolver ciertos problemas de forma más eficiente que los ordenadores clásicos. En computación, el término eficiente se utiliza para caracterizar algoritmos. Que un algoritmo sea eficiente significa que el tiempo que tarda ese algoritmo en resolver el problema para el que está diseñado, crece de forma polinómica con el tamaño de este problema [1]. Por ejemplo el tiempo de ejecución del algoritmo que se enseña en las escuelas para multiplicar dos números crece como n^2, siendo n el número de cifras de los dos números. Como el tiempo que se tarda en resolver un algoritmo depende también de la capacidad del ordenador que este ejecutándolo, se suele definir la eficiencia del algoritmo no en función del tiempo de ejecución, sino de la cantidad de pasos que necesita el algoritmo para ser ejecutado [2].

Para empezar a tratar las diferencias que existen entre la computación cuántica y la clásica es necesario tratar con más detalle las propiedades de la física cuántica que se introdujeron en el Capítulo 1. Para intentar entender las propiedades cuánticas se va a ampliar primero la notación de Dirac y después repasaremos brevemente los postulados de la mecánica cuántica.

2.1. Notación de Dirac

A continuación se va a ampliar la breve idea que se dio en el capítulo 1 sobre notación de Dirac. Aunque la notación de Dirac es mucho más amplia, en este trabajo solo se necesita trabajar con espacios discretos de dimensión finita, por lo tanto se explicará la notación solo para este caso. Para una explicación más completa de la notación se puede consultar el capítulo II del libro de Cohen-Tannoudji [3].

En mecánica cuántica el estado de un sistema está definido por una función de onda, que pertenece al conjunto de funciones de cuadrado integrable. Este conjunto tiene la estructura de un espacio de Hilbert, que es un espacio vectorial en los números complejos $\mathbb{C}$. A este espacio de estados lo llamamos $\mathscr{E}$. Cualquier elemento de este espacio vectorial se representa por $|\psi\rangle$ y se conoce por ket. El elemento correspondiente a $|\psi\rangle$ perteneciente al espacio dual[1], se le llama bra y se le representa por $\langle\psi|$. Podemos utilizar la notación $\langle\psi|\phi\rangle$ para representar al número que se obtiene al hacer el producto escalar de los dos kets $|\psi\rangle$ y $|\phi\rangle$. De

[1] El espacio dual del espacio $\mathscr{E}$ es el conjunto de aplicaciones lineales que asocian a cada $|\psi\rangle \in \mathscr{E}$ con un número complejo. Este espacio se representa como $\mathscr{E}^*$. [3].

este modo podemos imaginar los operadores lineales como objetos que asocian a cada ket con uno nuevo, se representan con una letra mayúscula.

En el caso de dimensión finita n, todos estos objetos los podemos representar como matrices en una cierta base ortonormal $\{|u_i\rangle\}$ que al ser base ortonormal cumple dos elementos distintos entre si de la base son ortonormales.

$$\langle u_i|u_j\rangle = \delta_{ij} \qquad i,j = 1,\cdots,n \tag{2.1}$$

y que la suma de los proyectores para cada elemento de la base suma la identidad

$$\sum_{i=1}^{n} |u_i\rangle\langle u_i| = \mathbb{1}. \tag{2.2}$$

En esta base podemos representar los kets como vectores columna:

$$|\psi\rangle = \begin{pmatrix} \psi_1 \\ \psi_2 \\ \vdots \\ \psi_n \end{pmatrix} = \sum_{i=1}^{n} \psi_i \, |u_i\rangle \,, \tag{2.3}$$

donde los números complejos ψ_i corresponden al número $\langle u_i|\psi\rangle$ que es la proyección de $|\psi\rangle$ sobre la base $|u_i\rangle$.
Los bras se representan como vectores fila:

$$\langle\psi| = \begin{pmatrix} \psi_1^* & \psi_2^* & \cdots & \psi_n^* \end{pmatrix} = \sum_{i=1}^{n} \psi_i^* \, \langle u_i| \,, \tag{2.4}$$

donde esta vez los números complejos ψ_i corresponden a $\langle u_i|\psi\rangle^* = \langle\psi|u_i\rangle$.

Los operadores lineales se escriben como matrices cuadradas donde cada elemento es $A_{ij} = \langle u_i| \, A \, |u_j\rangle$ y se agrupan de la siguiente forma:

$$A = \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1j} \\ A_{21} & A_{22} & \cdots & A_{2j} \\ \vdots & \vdots & \ddots & \vdots \\ A_{j1} & A_{j2} & \cdots & A_{jj} \end{pmatrix}. \tag{2.5}$$

Por lo tanto, para calcular la acción de un operador A sobre un ket $|\psi\rangle$, simplemente tendremos que multiplicar la matriz correspondiente a A por el vector columna correspondiente a $|\psi\rangle$, siempre y cuando todos los objetos mencionados estén expresados en la misma base.

2.2. Postulados

Una vez definida la notación utilizada en cuántica podemos hablar de cómo utilizar esta notación matemática para desarrollar la teoría física. Para ello vamos a enumerar los postulados en los que se cimienta la mecánica cuántica. Al igual que en la sección 2.1, se puede encontrar una explicación más extensa en el capítulo III del libro de Cohen-Tannooudji [3].

Postulado 1. En un tiempo concreto t_0, el estado de un sistema aislado está definido por su función de onda, que se corresponde con un ket $|\psi(t_0)\rangle$ perteneciente al espacio de estados $\mathscr{E}$.

Postulado 2. Toda cantidad física que se pueda medir viene descrita por un operador A que actúa en el espacio de estados. Este operador tiene la propiedad de ser un "observable", lo que significa que es un operador hermítico[2] y que sus vectores propios forman una base del espacio de estados.

Postulado 3. Los únicos resultados que se pueden obtener de una medida del observable A son los valores propios de ese observable.

Postulado 4. Si medimos el observable A sobre el estado $|\psi\rangle$, la probabilidad de obtener como resultado el valor propio a_n es: $P(a_n) = |\langle u_n|\psi\rangle|^2$, donde $|u_n\rangle$ es el estado propio normalizado del observable A correspondiente al autovalor a_n.

Postulado 5. Si después de una medida del observable A en el estado $|\psi\rangle$ se obtiene el resultado a_n, el estado inmediatamente después de la medida será:

$$|\psi'\rangle = \frac{P_n|\psi\rangle}{\sqrt{\langle\psi|P_n|\psi\rangle}},$$

es decir, la proyección normalizada de $|\psi\rangle$ en el subespacio de a_n, ya que $P_n = |u_n\rangle\langle u_n|$ es el proyector al subespacio $\mathscr{E}_n$ y $\langle\psi|P_n|\psi\rangle$ la probabilidad de medir a_n. El espacio $\mathscr{E}_n$ está generado por $|u_n\rangle$.

Postulado 6. La evolución temporal de un sistema viene dada por la ecuación de Schrödinger, que en la notación discreta se expresa así

$$i\hbar\frac{d}{dt}|\psi(t)\rangle = H(t)|\psi(t)\rangle,$$

donde $H(t)$ es el hamiltoniano del sistema, que es un observable.

Todos estos postulados se refieren al caso de un espectro discreto y no degenerado, que son los casos que vamos a tratar. Para el caso de un espectro degenerado habría que cambiar el postulado 4 por

$$P(a_n) = \sum_{i=1}^{g_n} |\langle u_n^i|\psi\rangle|^2, \tag{2.6}$$

donde g_n es el grado de degeneración del autovalor a_n y $\{|u_n^i\rangle\}$ un conjunto ortonormal de kets que son base del espacio asociado con el autovalor a_n. Dentro del postulado 5 habría que cambiar la definición de proyector sobre el subespacio $\mathscr{E}_n$ por

$$P_n = \sum_{i=1}^{g_n} |u_n^i\rangle\langle u_n^i|, \tag{2.7}$$

ya que ahora está generado por más de un vector.

2.3. Qubits versus bits

Una vez presentado el formalismo cuántico y sus postulados, podemos empezar a analizar las diferencias entre la computación cuántica y la clásica, que es de lo que trata esta sección. Para ello vamos a empezar a hablar de lo más básico: la unidad mínima de información. Como ya se introdujo en el capítulo 1 a la unidad mínima de información en computación clásica se la conoce como bit y en computación

[2] Un operador hermítico es aquel que es igual a su transpuesto conjugado.

cuántica como qubit. El qubit esencialmente es un vector en un espacio de Hilbert de dos dimensiones. La base más común de este espacio vectorial se suele denotar $\{|0\rangle, |1\rangle\}$ para mantener una similitud con la computación clásica. Estos vectores son los dados en la expresión (1.2). Por lo tanto el estado más general para un qubit será:

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\,. \tag{2.8}$$

La única restricción que tiene este estado es que esté normalizado, es decir que los números complejos α y β cumplan la condición

$$|\alpha|^2 + |\beta|^2 = 1, \tag{2.9}$$

lo cual dota a los qubits de mucha versatilidad. Es posible trabajar también con estados que no estén normalizados, siempre y cuando esto se tenga en cuenta a la hora de aplicar los postulados. En nuestro caso trabajaremos siempre con estados normalizados.

A pesar de que tenemos toda esta libertad para los qubits, a la hora de medir solo podemos obtener uno de los kets de la base en la que estemos midiendo, normalmente en el caso de un solo qubit utilizaremos la base $\{|0\rangle, |1\rangle\}$, por lo que podremos medir o bien $|0\rangle$ o bien $|1\rangle$ debido al postulado 4. Pero podemos operar con estados en superposición de los dos kets que es lo que hace tan potente la computación cuántica. La ventaja que presentan los qubits frente a los bits es la capacidad de cálculo. Imaginemos un ordenador con n bits, el estado de ese sistema será un único estado de los 2^n posibles (desde $00\cdots0$ hasta $11\cdots1$). Sin embargo, en un ordenador cuántico con n qubits el estado del sistema puede ser una combinación lineal de los 2^n posibles estados (desde $|00\cdots0\rangle$ hasta $|11\cdots1\rangle$) y podremos operar con esa combinación lineal de como máximo 2^n estados, aunque cuando midamos solo se obtenga uno de los posibles estados. Es importante mencionar también que en computación cuántica existe el fenómeno de interferencia. Este fenómeno se utiliza al operar con estados en superposición y se consigue que estos estados interfieran entre sí sacando así más partido al algoritmo.

Para poder visualizar estados de un solo qubit se puede utilizar la esfera de Bloch, capítulo 1 [1]. Para ello se pueden escribir los coeficientes α y β de tal forma que se les pueda dar una interpretación geométrica. Sabiendo que son números complejos es más conveniente escribirlo en la notación módulo-argumento. Tienen que cumplir la condición de estar normalizados (2.9) por lo que podemos interpretar esa parte real como el coseno y seno de un ángulo. Y ya que una fase global no afecta al sistema[3] , podemos poner la fase únicamente actuando sobre el ket $|1\rangle$. Utilizando esto podemos escribir la expresión (2.8) de la siguiente forma:

$$|\psi\rangle = \cos\frac{\theta}{2}\,|0\rangle + \sin\frac{\theta}{2}\,e^{i\varphi}\,|1\rangle\,, \tag{2.10}$$

con $\theta \in [0,\pi]$ y $\varphi \in [0,2\pi]$. Se pueden interpretar θ y φ como ángulos de las coordenadas esféricas y de esta forma representar los estados de un solo qubit como puntos sobre una esfera unitaria como se muestra en la figura 2.1.

Por ejemplo la representación de los estados $\frac{1}{\sqrt{2}}(|0\rangle \pm |1\rangle)$ sería la que se muestra en la figura 2.2.

Una vez expuestas las diferencias en la unidad básica de información entre un ordenador cuántico y uno clásico, vamos a hablar de las diferencias en los objetos que actúan sobre esas unidades básicas de información.

[3]Imaginemos un estado $|\psi'\rangle = e^{i\alpha}\,|\psi\rangle$ donde α es un número real. Si $|\psi\rangle$ está normalizado $|\psi'\rangle$ también lo está: $\langle\psi'|\psi'\rangle = \langle\psi|\,e^{-i\alpha}e^{i\alpha}\,|\psi\rangle = \langle\psi|\psi\rangle = 1$. A la hora de calcular probabilidades utilizando el postulado 4 $|\,\langle u_n|\psi'\rangle\,|^2 = |e^{i\alpha}\,\langle u_n|\psi\rangle\,|^2 = |\,\langle u_n|\psi\rangle\,|^2$ de modo que $|\psi'\rangle$ y $|\psi\rangle$ son el mismo estado [3].

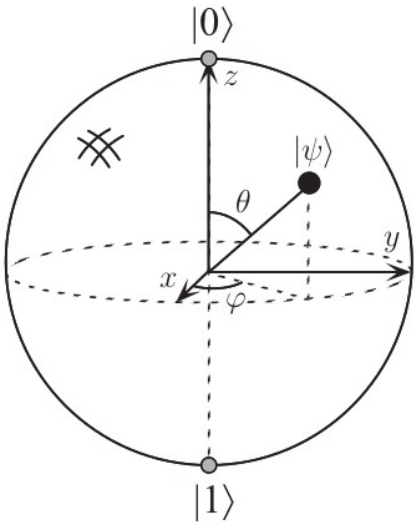


Figura 2.1: Representación esquemática de los ángulos θ y φ en la esfera de Bloch. Figura obtenida de [1].

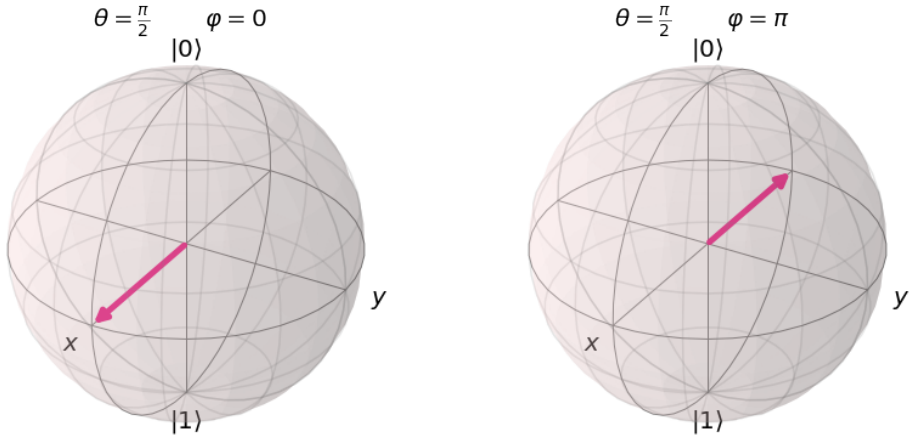
2.4. Puertas lógicas cuánticas

Una puerta lógica es un objeto que implementa una función lógica en un circuito. En computación cuántica las podemos imaginar como operadores que transforman el estado de un conjunto de qubits en otro. Este último estado debe de ser un estado válido. Podemos forzar que las puertas lógicas sean transformaciones que conserven la norma del estado, de tal forma que el estado final siga estando normalizado. Los operadores que preservan la norma de los estados son operadores unitarios, esto implica que siendo U un operador se cumpla que $UU^\dagger = \mathbb{1}$. De las leyes de la mecánica cuántica, que hemos comentado en la sección 2.2, se puede extraer que las transformaciones que se realicen sobre un estado tienen que ser reversibles. Esta es una propiedad que generalmente no está presente en computación clásica. La reversibilidad quiere decir que si a un estado inicial $|\psi\rangle$ le aplicamos una serie de puertas lógicas, siempre será posible obtener de nuevo el estado inicial $|\psi\rangle$. Esto último es verdad para todos los caso menos para el caso de hacer una medición en el sistema, ya que esa operación nos dará uno de los posibles resultados (lo que se conoce como colapso de la función de onda) y transformará el sistema de acuerdo con el postulado 5 de forma que nos será imposible saber cuál era el estado antes de la medida. Por lo tanto la medida no es una transformación unitaria ya que no tiene operación inversa.

Mientras que solo existen dos puertas lógicas que se puede aplicar a un solo bit, la puerta NOT que actúa cambiando el estado del bit $0 \to 1$ y $1 \to 0$ y la identidad que deja el bit inalterado, existen infinitas puertas lógicas que se pueden aplicar a un estado de un qubit. Esto sucede porque la única restricción que tienen las puertas lógicas es la de ser operaciones unitarias y, como vimos en la sección 2.1, los operadores se representan por matrices. En el caso de un solo qubit estas matrices son 2×2 y existen infinitas matrices unitarias de orden 2. Vamos a ver la representación de algunas de estas matrices en la base $\{|0\rangle, |1\rangle\}$ formada por los vectores introducidos en (1.2).

Hemos visto que la única puerta lógica que se puede aplicar a un estado de un solo bit clásico es la puerta NOT, que tiene su equivalente en cuántica y su representación matricial es:

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}. \tag{2.11}$$



(a) Estado $|+\rangle$ (b) Estado $|-\rangle$

Figura 2.2: Representación en la esfera de Bloch de los estados $|\pm\rangle = \frac{1}{\sqrt{2}}(|0\rangle \pm |1\rangle)$.

Podemos ver que la acción de este operador es cambiar el estado $|0\rangle$ por el $|1\rangle$ y viceversa y que la acción sobre un estado cualquiera como (2.8) es:

$$X|\psi\rangle = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \beta \\ \alpha \end{pmatrix}. \tag{2.12}$$

Es fácil de comprobar que $XX^\dagger = X^2 = \mathbb{1}$, y también que si el estado inicial esta normalizado ($|\alpha| + |\beta| = 1$) el estado final también lo está. También existen la puerta Y que se representa por:

$$Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \tag{2.13}$$

y la puerta Z representada por:

$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{2.14}$$

A estas tres puertas (X, Y y Z) se las conoce también como puertas de Pauli ya que su representación matricial coincide con la de las matrices de Pauli de orden 2.

Una de las puertas más importantes es la puerta H o de Hadamard. Esta puerta lo que hace es transformar los estados $|0\rangle$ y $|1\rangle$ en los estados $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ y $|-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$ respectivamente, que son los estados que están representados en la figura 2.2. La representación matricial de esta puerta es:

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}. \tag{2.15}$$

Este operador es importante ya que su acción es poner un qubit en superposición uniforme de los dos estados de la base $\{|0\rangle, |1\rangle\}$, o visto de otra forma, hacer un cambio de base a la dada por los estados $\{|+\rangle, |-\rangle\}$.

La puerta llamada "fase"nos permite dotar al estado $|1\rangle$ de una fase con respecto al estado $|0\rangle$, la representación matricial de esta puerta es

$$P(\lambda) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\lambda} \end{pmatrix}, \tag{2.16}$$

y podemos observar que en el caso de que $\lambda = \pi$ la puerta fase se convierte en la puerta Z que hemos visto en (2.14). Existen también más valores de λ que dan lugar a puertas con nombre propio, por ejemplo la puerta T con $\lambda = \pi/4$ y la puerta S con $\lambda = \pi/2$. Estas puertas solo tendrán relevancia en el caso de ser aplicadas sobre un estado con más de un ket. Si actúan sobre $|0\rangle$ o sobre $|1\rangle$ únicamente, dotarán a este elemento de una fase global, que como hemos visto no tiene efecto neto, pie de página 3. Se podría seguir mencionando posibles puertas lógicas cuánticas, pero al tener como única condición el ser unitarias existen infinitas. Por lo tanto se van a comentar puertas lógicas cuánticas que actúan sobre más de un qubit.

2.4.1. Puertas lógicas de varios qubits

También existen puertas que actúan sobre múltiples qubits a la vez, al igual que en el caso de un solo qubit existen infinitas, por ello vamos a comentar una particular y luego un grupo de puertas lógicas.

A lo largo de esta sección las matrices expuestas están dadas en la base $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$

$$|00\rangle = \begin{pmatrix}1\\0\\0\\0\end{pmatrix} \quad |01\rangle = \begin{pmatrix}0\\1\\0\\0\end{pmatrix} \quad |10\rangle = \begin{pmatrix}0\\0\\1\\0\end{pmatrix} \quad |11\rangle = \begin{pmatrix}0\\0\\0\\1\end{pmatrix}. \tag{2.17}$$

La primera puerta que vamos a comentar es la puerta denominada $SWAP$ que permite intercambiar el estado de dos qubits. Este operador, al actuar sobre dos qubits, está representado por una matriz cuadrada de orden 4:

$$SWAP = \begin{pmatrix}1&0&0&0\\0&0&1&0\\0&1&0&0\\0&0&0&1\end{pmatrix}. \tag{2.18}$$

Por lo tanto si se quiere ver la acción de esta puerta sobre un estado general de cuatro qubits $|\chi\rangle = \alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \delta|11\rangle$, donde $|\alpha|^2 + |\beta|^2 + |\gamma|^2 + |\delta|^2 = 1$, habrá que hacer la multiplicación de la matriz (2.18) por el vector correspondiente a $|\chi\rangle$.

$$SWAP|\chi\rangle = \begin{pmatrix}1&0&0&0\\0&0&1&0\\0&1&0&0\\0&0&0&1\end{pmatrix}\begin{pmatrix}\alpha\\\beta\\\gamma\\\delta\end{pmatrix} = \begin{pmatrix}\alpha\\\gamma\\\beta\\\delta\end{pmatrix}, \tag{2.19}$$

es decir, intercambia los estados de los qubits 1 y 2 ($|a,b\rangle \to |b,a\rangle$).

Ahora pasamos a comentar un conjunto muy importante de puertas lógicas de varios qubits, que es el conjunto de las puertas controladas. Estas puertas se caracterizan por tener un grupo de qubits de control y otro grupo como objetivo. Para explicarlo vamos a poner el ejemplo de la puerta para varios qubits más conocida, la $CNOT$. Esta puerta tiene un qubit control y un qubit objetivo y la operación que hace es $|a,b\rangle \to |a, a \oplus b\rangle$, donde $\oplus$ es la suma módulo 2. La acción de esta puerta se puede entender como una operación condicional: si el qubit control está en el estado $|1\rangle$, aplica la puerta X sobre el estado objetivo. La representación matricial de esta puerta en el caso de que el qubit control sea el primero y el qubit objetivo el segundo es:

$$CNOT = \begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{pmatrix}. \tag{2.20}$$

Un ejemplo para ver la utilidad de esta puerta es la creación del estado entrelazado

$$\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle). \tag{2.21}$$

Para ello se empieza con dos qubits en el estado $|00\rangle$. Al primero de ellos se le aplica una puerta Hadamard creando así la superposición uniforme $1/\sqrt{2}(|00\rangle + |10\rangle)$. A continuación se aplica una puerta $CNOT$ con el primer qubit como control y el segundo como objetivo. De esta forma, como en el primer sumando el qubit control está en el estado $|0\rangle$ no se cambia el estado del segundo qubit, pero en el segundo sumando el qubit control sí que está en el estado $|1\rangle$, por lo que habrá que aplicar la puerta X en el segundo registro. En resumen:

$$|00\rangle \xrightarrow{H} \frac{1}{\sqrt{2}}(|00\rangle + |10\rangle) \xrightarrow{CNOT} \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle). \tag{2.22}$$

Una vez entendida la puerta $CNOT$ podemos imaginarnos que es posible hacer todas las puertas como una puerta controlada, es decir, cualquier puerta que se pueda aplicar sobre un solo qubit la podemos poner con un qubit de control consiguiendo así que solo se aplique la puerta lógica si el qubit de control está en el estado $|1\rangle$. Este hecho lo podemos extender a más qubits haciendo que haya más de un qubit de control y que no se aplique el operador a no ser que todos los qubits estén en el estado $|1\rangle$. También se puede extender l idea a puertas que actúen sobre dos qubits directamente, como por ejemplo a la puerta $SWAP$ añadirle un qubit control convirtiéndola así en una puerta de tres qubits.

Capítulo 3

El algoritmo de Grover

En este capítulo vamos a tratar el algoritmo de Grover. Este algoritmo fue introducido en 1997 [4] y lleva el nombre de su autor. Es junto, con el de Shor [5], uno de los primeros ejemplos de algoritmos cuánticos que pueden resolver un problema de forma más eficiente que su análogo clásico. Grover ideó un algoritmo que permite encontrar de forma más eficiente elementos de entre un conjunto de datos. Mientras que el algoritmo de Shor presenta una ventaja exponencial frente al mejor algoritmo clásico conocido, el de Grover solo presenta una ventaja cuadrática, es decir, un algoritmo clásico completará el proceso de búsqueda en $O(N)$ pasos a diferencia del cuántico que lo hará en $O(\sqrt{N})$. Vamos a empezar dando una motivación física aproximada del algoritmo de Grover y luego en la sección 3.2 nos abstraeremos para explicar la generalización del algoritmo como se hace en [4].

3.1. Pozo de potencial

Lo que se intenta hacer es encontrar un estado de entre un conjunto de N estados. Para ello se hará evolucionar el sistema gradualmente de tal forma que en cada evolución elemental el estado deseado gane amplitud y el resto la pierdan, de tal forma que tras un cierto tiempo la amplitud del estado marcado será la suficiente como para que, al medir el estado general, se encuentre este elemento con una probabilidad grande. Para ello se puede empezar con una superposición uniforme de todos los estados y someterla a un potencial concreto, de tal forma que los estados evolucionen según ese potencial hacia el estado deseado como se muestra en la fígura 3.1 Por lo tanto, dotando al estado deseado de un potencial más bajo se podría conseguir que, dando suficiente tiempo al sistema, el estado deseado obtenga una amplitud suficiente como para medirlo con seguridad. Esta idea se puede encontrar en un artículo escrito por Grover [6].

En el trabajo que acabo de mencionar se parte de la ecuación de Schrödinger en presencia de un potencial que, salvo ciertas constantes, es:

$$\frac{\partial}{\partial t}\psi(x,t) = i\frac{\partial^2}{\partial x^2}\psi(x,t) - iV(x)\psi(x,t). \tag{3.1}$$

Aquí es útil considerar el espacio unidimensional como una línea compuesta por pequeños segmentos de longitud dx, de esta forma el espacio esta discretizado. A su vez también es útil dividir la evolución temporal en pequeños intervalos dt. Si consideramos esto último y la definición usual como cociente incremental de las derivadas

$$\frac{\partial}{\partial t}\psi(x,t) = \frac{\psi(x,t+dt) - \psi(x,t)}{dt}, \qquad \frac{\partial^2}{\partial x^2}\psi(x,t) = \frac{\psi(x+dx,t) + \psi(x-dx,t) - 2\psi(x,t)}{(dx)^2}, \tag{3.2}$$

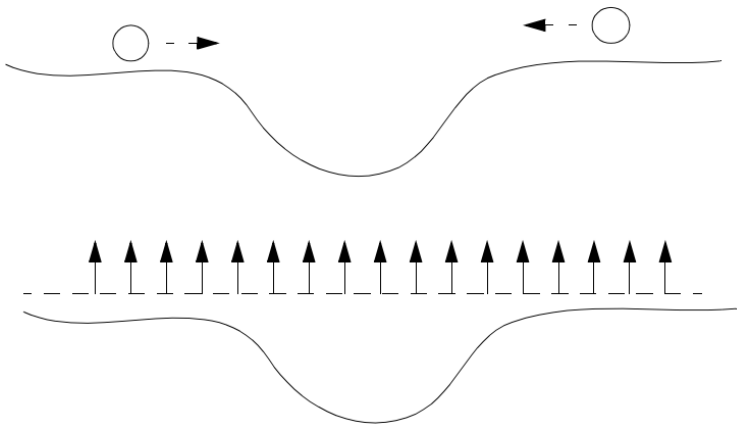


Figura 3.1: Símil entre dos bolas que evolucionan hacia puntos con menor energía potencial y la evolución del algoritmo de Grover mediante un potencial. Figura obtenida de [6].

se llega a que podemos expresar la ecuación (3.1) de una forma más conveniente como

$$\psi(x,t+dt) = (1 - iV(x) - 2i\varepsilon)\psi(x,t) + i\varepsilon\psi(x+dx,t) + i\varepsilon\psi(x-dx,t), \tag{3.3}$$

donde se ha hecho el cambio $dx = \sqrt{dt/\varepsilon}$. Esta ecuación expresa cómo afectan los estados más próximos a la evolución temporal de un estado dado. La expresión (3.3) se puede poner en forma matricial si se considera un estado finito y discreto. Por ejemplo, para cuatro estados los estados 4^{o} y 1^{o} están conectados y quedan condiciones de contorno circulares que se traducen en una matriz:

$$\vec{\psi}(\vec{x},t+dt) = \begin{pmatrix} 1 - iV(x_1)dt - 2i\varepsilon & i\varepsilon & 0 & i\varepsilon \\ i\varepsilon & 1 - iV(x_2)dt - 2i\varepsilon & i\varepsilon & 0 \\ 0 & i\varepsilon & 1 - iV(x_3)dt - 2i\varepsilon & i\varepsilon \\ i\varepsilon & 0 & i\varepsilon & 1 - iV(x_4)dt - 2i\varepsilon \end{pmatrix} \vec{\psi}(\vec{x},t). \tag{3.4}$$

La matriz que se obtiene se puede expresar como multiplicación de dos matrices siempre y cuando se desprecien los términos de orden dt^2, ε^2 y εdt y considerando que $V(x)$ es lo suficientemente pequeño para hacer la aproximación $e^{-iV(x)dt} = 1 - iV(x)dt$. Estas dos matrices son, para el caso de cuatro estados,

$$D = \begin{pmatrix} 1-2i\varepsilon & i\varepsilon & 0 & i\varepsilon \\ i\varepsilon & 1-2i\varepsilon & i\varepsilon & 0 \\ 0 & i\varepsilon & 1-2i\varepsilon & i\varepsilon \\ i\varepsilon & 0 & i\varepsilon & 1-2i\varepsilon \end{pmatrix} \quad \text{y} \quad R = \begin{pmatrix} e^{-iV(x_1)dt} & 0 & 0 & 0 \\ 0 & e^{-iV(x_2)dt} & 0 & 0 \\ 0 & 0 & e^{-iV(x_3)dt} & 0 \\ 0 & 0 & 0 & e^{-iV(x_4)dt} \end{pmatrix}. \tag{3.5}$$

Se puede comprobar que si se hace la multiplicación de estas dos matrices $(D \cdot R)$ y se realizan las aproximaciones ya mencionadas se vuelve a obtener la matriz (3.4).

Por lo tanto, se han obtenido dos matrices que nos dan la evolución temporal infinitesimal de un estado en función del potencial al que esté sometido. Si nos fijamos en la matriz D es fácil ver que la suma de los elementos de una columna es la unidad, por lo tanto, podemos considerar que D representa una matriz de transición para un proceso de Markov. Un proceso de Markov es aquel en el que el estado del sistema se mueve entre varios estados posibles con una probabilidad de transición que solo depende el estado actual y del estado destino. No hay ninguna dependencia de la probabilidad en los estados anteriores. Se puede

decir que es un proceso sin memoria. Por lo tanto cada uno de los elementos de la matriz representa la amplitud de probabilidad de transición entre estados. Si fijamos una fila, la suma de todos los elementos tiene que ser 1.

Volviendo al principio de esta sección, comentábamos que se podría iniciar una superposición uniforme de todos los estados posibles y someter al estado que nos interesa a un potencial menor, de tal forma que el sistema evolucionase hacia el estado que nos interesa. Por lo tanto podemos definir el potencial que necesitamos del siguiente modo,

$$V(x) = \begin{cases} -\gamma & \text{si} \quad x \in S, \\ 0 & \text{si} \quad x \notin S, \end{cases} \qquad \gamma > 0, \tag{3.6}$$

donde S es el conjunto de los estados que nos interesan. De momento vamos a trabajar con un solo estado perteneciente a S, pero podría haber más. Se ha definido el potencial al que hay que someter a la superposición uniforme de todos los estados y, si sustituimos este potencial en la matriz R, vemos que su efecto es dotar al elemento seleccionado de una fase. Si suponemos como elemento marcado el segundo, la matriz R es

$$R = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & e^{i\gamma} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{3.7}$$

Como se ha dicho, la matriz D puede considerarse como matriz de transición de un proceso de Markov y cada elemento representa la amplitud de probabilidad entre dos estados. La matriz representada en (3.5) tiene ceros, lo que implica que la probabilidad de transición entre algunos pares de estados es nula. Se puede sacar más partido de la matriz si consideramos un proceso de difusión que afecte a todo el pozo, de tal forma que se pueden conectar estados que no estaban conectados en (3.5). La matriz D quedaría de la siguiente forma:

$$D = \begin{pmatrix} 1-3i\varepsilon & i\varepsilon & i\varepsilon & i\varepsilon \\ i\varepsilon & 1-3i\varepsilon & i\varepsilon & i\varepsilon \\ i\varepsilon & i\varepsilon & 1-3i\varepsilon & i\varepsilon \\ i\varepsilon & i\varepsilon & i\varepsilon & 1-3i\varepsilon \end{pmatrix}. \tag{3.8}$$

Se puede comprobar que si sumamos los elementos de las filas siguen resultando 1. Ahora queda por determinar cuántas veces hay que aplicar la operación $D \cdot R$ al estado inicial para conseguir el estado que nos interesa, cuál es el valor de ε y cuál es el valor de γ.

Para calcular todos estos valores vamos a considerar un estado de N elementos y si se toma N lo suficientemente grande se pueden hacer aproximaciones que simplifican los cálculos. La matriz R para un espacio de dimensión N difiere de (3.7) en el número de filas y columnas únicamente, pero en la matriz D aparte de variar el número de filas y columnas, cambian también los elementos de la diagonal.

$$D = \begin{pmatrix} 1-Ni\varepsilon & i\varepsilon & i\varepsilon & \cdots & i\varepsilon \\ i\varepsilon & 1-Ni\varepsilon & i\varepsilon & \cdots & i\varepsilon \\ i\varepsilon & i\varepsilon & 1-Ni\varepsilon & \cdots & i\varepsilon \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ i\varepsilon & i\varepsilon & i\varepsilon & \cdots & 1-Ni\varepsilon \end{pmatrix}. \tag{3.9}$$

En esta matriz la suma de los elementos de una columna no es la unidad $(1 - i\varepsilon)$ pero si consideramos N lo suficientemente grande el error que se comete es mínimo.

Ya se ha comentado el efecto de la matriz R, que dota al estado objetivo de una fase γ con respecto al resto de estados, y el efecto de D que es intercambiar amplitudes entre estados. Teniendo esto en cuenta

podemos ver que entre los estados que no estén rotados el intercambio neto de amplitud de probabilidad va a ser 0, ya que la amplitud que cederán a los estados no marcados les será devuelta por esos estados. Por lo tanto, solo hay intercambio de amplitud entre el estado marcado y el resto.

Una vez comentado el efecto de las matrices vamos a comprobarlo con un estado. Iniciamos con una superposición uniforme de todos los estados, donde los estados no marcados tienen amplitud $k/\sqrt{N}$ y el marcado $K/\sqrt{N}$. Entonces:

$$\psi = \frac{1}{\sqrt{N}}\begin{pmatrix} k & K & k & \cdots & k \end{pmatrix} \xrightarrow{R(\frac{\pi}{2})} \frac{1}{\sqrt{N}}\begin{pmatrix} k & iK & k & \cdots & k \end{pmatrix} \tag{3.10}$$

si suponemos que $\gamma = \pi/2$. Ahora aplicamos el operador D sobre el estado obtenido al final de (3.10) y se obtiene

$$\frac{1}{\sqrt{N}}\begin{pmatrix} 1-Ni\varepsilon & i\varepsilon & i\varepsilon & \cdots & i\varepsilon \\ i\varepsilon & 1-Ni\varepsilon & i\varepsilon & \cdots & i\varepsilon \\ i\varepsilon & i\varepsilon & 1-Ni\varepsilon & \cdots & i\varepsilon \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ i\varepsilon & i\varepsilon & i\varepsilon & \cdots & 1-Ni\varepsilon \end{pmatrix}\begin{pmatrix} k \\ iK \\ k \\ \vdots \\ k \end{pmatrix} = \frac{1}{\sqrt{N}}\begin{pmatrix} (1-iN\varepsilon)k - K\varepsilon + (N-2)ki\varepsilon \\ (N-1)ki\varepsilon + Ki + KN\varepsilon \\ (1-iN\varepsilon)k - K\varepsilon + (N-2)ki\varepsilon \\ \vdots \\ (1-iN\varepsilon)k - K\varepsilon + (N-2)ki\varepsilon \end{pmatrix}, \tag{3.11}$$

donde podemos hacer la aproximación $N-1 \approx N$ y $N-2 \approx N$ para que al final nos quede una amplitud para el estado marcado que es

$$\frac{iK}{\sqrt{N}} + \frac{ik}{\sqrt{N}}N\varepsilon + \frac{K}{\sqrt{N}}N\varepsilon, \tag{3.12}$$

y para los no marcados

$$\frac{k}{\sqrt{N}} - \frac{K\varepsilon}{\sqrt{N}} - \frac{ik\varepsilon}{\sqrt{N}}. \tag{3.13}$$

Con estas amplitudes podemos ver cómo varía la amplitud y la fase de los estados con cada aplicación de D y R. En (3.13) podemos ver que la amplitud y la fase de los estados no marcados no varía prácticamente, ya que ε tiene un valor pequeño. En cuanto al estado marcado podemos escribir su amplitud como

$$\frac{iK}{\sqrt{N}}(1-iN\varepsilon) + \frac{ikN\varepsilon}{\sqrt{N}} \simeq \frac{iK}{\sqrt{N}}e^{-iN\varepsilon} + \frac{ikN\varepsilon}{\sqrt{N}}. \tag{3.14}$$

Con esta nueva amplitud podemos calcular cómo ha variado la probabilidad de encontrar el estado de forma aproximada. Para calcular el desfase que se ha añadido al nuevo estado, teniendo en cuenta que $N\varepsilon$ es algo pequeño, podemos despreciar el segundo término de (3.14) de tal forma que nos queda $\frac{iK}{\sqrt{N}}e^{-iN\varepsilon}$, que es la amplitud antes de aplicar el operador D más una fase igual a $-N\varepsilon$, que es una fase pequeña. Para calcular cómo varía la amplitud tenemos que considerar que el desfase es pequeño porque podemos suponer $e^{-iN\varepsilon} \simeq 1$. Al suponer esto nos queda un número imaginario puro cuyo módulo es $\frac{K}{\sqrt{N}} + \frac{kN\varepsilon}{\sqrt{N}}$, que es la amplitud original más otro término. Sabiendo que k es de orden 1 podemos ver que, aproximadamente, la amplitud del estado marcado aumenta en $\frac{N\varepsilon}{\sqrt{N}}$ cada iteración.

Si nos fijamos ahora en (3.13) podemos ver que la amplitud y la fase de los estados no marcados varía de forma despreciable. Esto viene dado por que todos los estados no marcados dan un poco de su amplitud al estado marcado y al ser el número de estados no marcados un número grande el efecto sí que es apreciable en el estado objetivo, mientras que para el resto de estados se puede desechar.

En conclusión, con cada aplicación de la operación $D \cdot R$ la amplitud del estado marcado aumenta en aproximadamente $\frac{N\varepsilon}{\sqrt{N}}$ y su fase cambia un valor de $N\varepsilon$ con respecto a la fase que tenía después de la primera aplicación de R (3.10), mientras que los estados no marcados no varían ni su fase ni su amplitud

dentro de las aproximaciones realizadas. El cálculo que hemos realizado solo tiene validez cuando el estado marcado tiene una fase de $\pi/2$ con respecto al resto de estados, después de la primera aplicación de $D \cdot R$ el estado marcado presenta una fase de $\pi/2 - N\varepsilon$ por lo tanto tendremos que preparar la matriz R para que dote al estado marcado de una fase $N\varepsilon$ para que cuando se vuelva a aplicar la matriz D ocurra el mismo proceso.

Se observa que cuanto más grande sea ε más grande es la transferencia de amplitud entre estados, y antes se consigue que el estado marcado tenga una amplitud suficiente como para que al medir se obtenga este estado con mucha seguridad. Pero tenemos restricciones a lo grande que puede ser ε ya que es necesario que la matriz D (3.9) sea unitaria. Sabemos que las columnas de una matriz unitaria forman una base ortonormal del espacio, y el producto escalar de dos columnas nos da $O(N\varepsilon^2)$ y el módulo de una columna es $1 + O(N^2\varepsilon^2)$, por lo tanto la matriz D será unitaria siempre y cuando se puedan despreciar estos términos y tengamos que el producto escalar de dos columnas sea 0 y el módulo de una columna sea 1. Viendo que la amplitud que se gana en cada iteración es $\frac{N\varepsilon}{\sqrt{N}}$ si se considera que $\varepsilon = O(\frac{1}{N})$ se obtiene que la transferencia de amplitud en cada paso es $O(\frac{1}{\sqrt{N}})$, por ello después de $O(\sqrt{N})$ repeticiones del paso del algoritmo encontraremos que el estado marcado tiene una amplitud de orden 1, que es lo que necesitábamos, de aquí se obtiene ese valor que presentamos al inicio de esta sección y el porqué este algoritmo es más eficiente que el clásico.

3.2. Generalización del algoritmo

En la sección anterior hemos mostrado cómo se podía aumentar la probabilidad de medir un estado mediante transferencias de amplitud. Aunque la idea es aproximada nos sirve para sentar las bases de los pasos necesarios para ejecutar el algoritmo. Al igual que antes será necesario una rotación de fase y una transferencia de amplitud desde los estados que no nos interesan hasta el elemento marcado. En [4] podemos ver qué operaciones son las que consiguen aumentar la amplitud. Para empezar es necesario conseguir como estado inicial una superposición uniforme de todos los estados posibles. La transformación que nos consigue esto es la puerta Hadamard (2.15) extendida a n qubits

$$H^{\otimes n} = H \otimes H \otimes \cdots \otimes H, \tag{3.15}$$

que es una matriz de dimensión $2^n \times 2^n = N \times N$. Si se aplica esta puerta al estado $|00\cdots0\rangle$ se consigue una superposición uniforme de los 2^n posibles estados en fase. La siguiente transformación que necesitamos se conoce como transformación de difusión D, que es la equivalente a la matriz D de la sección 3.1. Los elementos de matriz se definen como:

$$D_{ij} = -\delta_{ij} + \frac{2}{N}, \tag{3.16}$$

dando lugar a

$$D = \begin{pmatrix} -1+\frac{2}{N} & \frac{2}{N} & \cdots & \frac{2}{N} \\ \frac{2}{N} & -1+\frac{2}{N} & \cdots & \frac{2}{N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{2}{N} & \frac{2}{N} & \cdots & -1+\frac{2}{N} \end{pmatrix}. \tag{3.17}$$

La matriz (3.17) se puede expresar como $D = -\mathbb{1} + 2P$, donde la matriz P se define como $P_{ij} = 1/N$ y es un proyector. Para que una puerta lógica cuántica sea válida tiene que ser unitaria $U^\dagger U = \mathbb{1}$, donde $U^\dagger$ es la transpuesta conjugada. Como D es simétrica y real, es su propia transpuesta conjugada, $D^\dagger = D$, por lo tanto tendremos que comprobar si $D^2 = \mathbb{1}$. Efectuando el cálculo

$$(-\mathbb{1} + 2P)^2 = 4P^2 + \mathbb{1} - 4P,$$

pero al ser P un proyector $P^2 = P$, en efecto

$$\begin{pmatrix} \frac{1}{N} & \frac{1}{N} & \cdots & \frac{1}{N} \\ \frac{1}{N} & \frac{1}{N} & \cdots & \frac{1}{N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{N} & \frac{1}{N} & \cdots & \frac{1}{N} \end{pmatrix} \cdot \begin{pmatrix} \frac{1}{N} & \frac{1}{N} & \cdots & \frac{1}{N} \\ \frac{1}{N} & \frac{1}{N} & \cdots & \frac{1}{N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{N} & \frac{1}{N} & \cdots & \frac{1}{N} \end{pmatrix} = \begin{pmatrix} \frac{1}{N} & \frac{1}{N} & \cdots & \frac{1}{N} \\ \frac{1}{N} & \frac{1}{N} & \cdots & \frac{1}{N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{N} & \frac{1}{N} & \cdots & \frac{1}{N} \end{pmatrix}.$$

En conclusión $D^2 = \mathbb{1}$, por lo que la operación difusión se puede considerar una puerta lógica válida.

El operador P cuando actúa sobre un vector nos devuelve otro vector donde cada una de sus componentes corresponde a la media de todas las componentes del vector original. Es por eso que al operador D también se le conoce como inversión sobre la media, ya que cuando actúa sobre un vector cualquiera nos devuelve otro vector donde las componentes son el simétrico de las originales con respecto a la media. Por consiguiente si una componente se encuentra muy cerca de la media después de la operación de difusión apenas habrá variado su valor, mientras que si una está alejada de la media su valor después habrá cambiado. Esto se puede ver claramente en la figura 3.2. La primera componente del vector coincide con la media por lo tanto no se ve afectada por el operador D; la segunda tiene un valor próximo a la media por lo que varía poco su valor y por ultimo las componentes finales están más distanciadas del valor de la media y su desplazamiento es mayor.

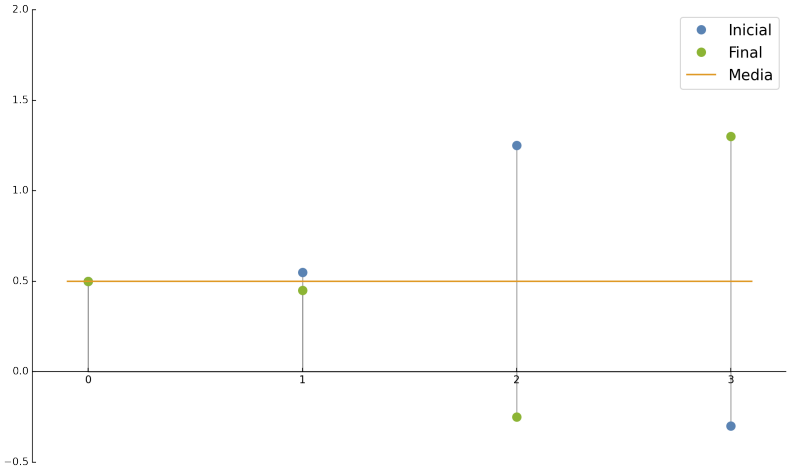


Figura 3.2: Representación de la acción del operador D sobre las componentes de un vector en el caso $N = 5$.

En conclusión, después de que sobre un vector $\vec{v}$ haya actuado el operador D, cada una de las componentes adquiere un nuevo valor que viene dado por

$$M + (M - v_i), \tag{3.18}$$

donde M es el valor medio de todas las componentes del vector $\vec{v}$ y v_i es el valor de la componente i.

Una vez definidas las operaciones que vamos a utilizar, tenemos que mostrar qué pasos sigue este algoritmo para encontrar el estado deseado. Se empieza aplicando la puerta Hadamard (3.15) al estado

$|00\cdots0\rangle$, lo que consigue la superposición de todos los N posibles estados

$$|\phi_0\rangle = H^{\otimes n}\,|00\cdots0\rangle = \frac{1}{\sqrt{N}}\sum_{x=0}^{2^n}|x\rangle \tag{3.19}$$

con amplitud $1/\sqrt{N}$. Ahora se aplica una rotación de fase de π radianes al estado marcado de tal forma que nos quede con la misma amplitud inicial $(1/\sqrt{N})$ pero negativa

$$R\,|\phi_0\rangle = \frac{1}{\sqrt{N}}\,|1,1,\cdots,1,-1,1,\cdots,1\rangle \tag{3.20}$$

Si nos fijamos en este estado y dado que suponemos N un número grande, podemos considerar que la media de todas las componentes es $1/\sqrt{N}$ y no cometeremos apenas error. Por ende, si se aplica ahora la operación (3.17) al estado (3.20) y dado que el elemento marcado tiene una amplitud negativa y por tanto alejada de la media, se obtendrá un nuevo estado en el que la amplitud del marcado ha crecido con respecto a la del resto de estados. La amplitud del resto de elementos del estado después de aplicar la rotación de fase es $1/\sqrt{N}$, que realmente es algo superior a la media de todos los estados, por lo que los elementos no marcados en $|\phi_1\rangle = DR\,|\phi_0\rangle$ tendrán una amplitud menor de $1/\sqrt{N}$ y habrá ocurrido la transferencia de amplitud de los elementos no marcados hacia el objetivo, como se comentó en la sección 3.1.

En la figura 3.3 podemos ver los pasos que seguimos con el algoritmo de Grover. Empezamos con una superposición uniforme

$$|\phi_0\rangle = \frac{1}{\sqrt{N}}\sum_{x=0}^{N}|x\rangle \tag{3.21}$$

que está representada en la figura 3.3a. A este estado le aplicamos la puerta R, aplicando así una fase de π al estado marcado. Es por eso que la amplitud del estado marcado en la figura 3.3b tiene signo negativo. Invertimos sobre la media haciendo así que el estado marcado aumente su amplitud y el resto de elementos pierdan, como se ve en la figura 3.3c.

Para ver como varia la amplitud de los estados con cada iteración se puede usar la misma estrategia que se ha seguido en la anterior sección en las ecuaciones (3.10), (3.11), (3.12) y (3.13). Denotamos la amplitud de los estados no marcados como $C/\sqrt{N}$, donde C es una constante con valor $\frac{1}{2} \leq C < 1$. Con este valor el estado marcado tiene que tener una amplitud

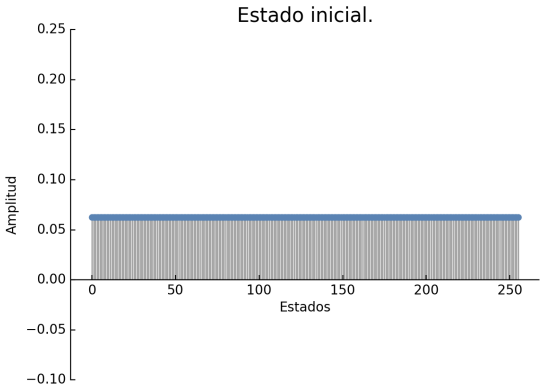
$$\sqrt{1-\left(\frac{C}{\sqrt{N}}\right)^2(N-1)} \simeq \sqrt{1-\left(\frac{C}{\sqrt{N}}\right)^2 N} = \sqrt{1-C^2}.$$

La media la podemos considerar $C/\sqrt{N}$ sin cometer apenas error, así pues los elementos no marcados variarán muy poco su amplitud. Si ahora aplicamos las puertas R y D a este estado observamos que el elemento con amplitud distinta se transforma en

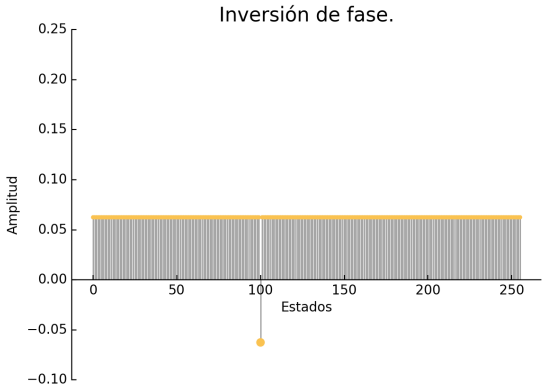
$$\sqrt{1-C^2} - \frac{2}{N}\sqrt{1-C^2} + (N-1)\frac{2C}{N\sqrt{N}}. \tag{3.22}$$

Podemos ver que la nueva amplitud está compuesta de varios términos, el primero es la amplitud que tenia inicialmente, el segundo es la amplitud que pierde y es transferida a los estados no marcados y el último término es la amplitud que gana de cada estado no marcado, de ahí que esté multiplicado por $N-1$. Si en (3.22) desarrollamos y despreciamos los términos $O(N^{-1})$ conseguimos llegar a

$$\sqrt{1-C^2} + \frac{2C}{\sqrt{N}}. \tag{3.23}$$



(a) $|\phi_0\rangle$



(b) $R\,|\phi_0\rangle$

(c) $|\phi_1\rangle$

Figura 3.3: Representación gráfica del primer paso del algoritmo de Grover.

Se puede ver una similitud entre esta expresión y la dada en (3.18), la media en este caso vale $C/\sqrt{N}$ y el valor de la componente marcada es $-\sqrt{1-C^2}$ antes de aplicar D. Con ello podemos observar que la amplitud en cada iteración aumenta en $\frac{2C}{\sqrt{N}}$, que equivale al valor $O(\frac{1}{\sqrt{N}})$ que se calculó en la sección 3.1. En consecuencia, en $O(\sqrt{N})$ iteraciones se conseguirá una amplitud lo suficientemente grande. Además se puede ver que si $C > 1/\sqrt{2}$ la ganancia en amplitud será mayor que $\frac{1}{\sqrt{2N}}$, por lo que debe existir un número $m < \sqrt{N}$ de repeticiones del algoritmo tras el que la amplitud del estado marcado sea mayor que $1/\sqrt{2}$ y por lo tanto la probabilidad de medirlo sea mayor del 50 %.

Hasta ahora se han comentado aproximaciones para encontrar el número de pasos necesarios para llegar a una solución fiable, pero es importante intentar dar una respuesta más concreta ya que la amplitud del estado marcado no crece de forma indefinida, sino que vuelve a caer. Por ejemplo, después de $\sqrt{2N}$ iteraciones la probabilidad de medir el estado marcado es menor del 9,5 %. En el artículo [7] aportan estos datos y además se da un cálculo para el número de iteraciones necesarias. En este artículo se trabaja con

el estado

$$|\Phi(K,k)\rangle = K\,|i_0\rangle + \sum_{i\neq i_0} k\,|i\rangle \tag{3.24}$$

donde $|i_0\rangle$ es el elemento marcado y K la amplitud que tiene. Nuestro objetivo será ver cómo varía la amplitud K en cada iteración. Para ello comenzamos con la superposición uniforme de todos los estados $|\phi_0\rangle = |\Phi(1/\sqrt{N}, 1/\sqrt{N})\rangle$ y aplicamos el algoritmo un número de veces. Se puede llegar a una fórmula recursiva para las amplitudes, que será, tras j iteraciones

$$\begin{cases} K_{j+1} = \frac{N-2}{N}K_j + \frac{2(N-1)}{N}k_j, \\ \\ k_{j+1} = \frac{N-2}{N}k_j - \frac{2}{N}K_j, \end{cases} \tag{3.25}$$

que se puede llegar a transformar en una expresión más compacta y en la que la dependencia entre k_j y K_j no se muestre de forma explícita, en concreto.

$$\begin{cases} K_j = \sin((2j+1)\theta), \\ \\ k_j = \frac{1}{\sqrt{N-1}}\cos((2j+1)\theta), \end{cases} \tag{3.26}$$

donde el ángulo θ se define como $\sin^2\theta = 1/N$. Ahora vamos a demostrar que tras $m = \lfloor \pi/4\theta \rfloor$ iteraciones la probabilidad de encontrar el estado que nos interesa es suficientemente grande.
Sabiendo que las amplitudes en cada paso vienen dadas por (3.26) se puede ver que la mayor probabilidad de encontrar el estado marcado vendrá dado cuando k_j sea muy próximo a cero. Llamamos $\tilde{m}$ al número que cumple $k_{\tilde{m}} = 0$ y esto sucede cuando $(2\tilde{m}+1)\theta = \frac{\pi}{2} \to \tilde{m} = (\pi - 2\theta)/4\theta$. Ahora nombramos a $m = \lfloor \pi/4\theta \rfloor$. Con estas definiciones de m y $\tilde{m}$ se comprueba que $|m - \tilde{m}| = |\lfloor \frac{\pi}{4\theta} \rfloor - \frac{\pi}{4\theta} + \frac{1}{2}| \leq \frac{1}{2}$. También existe la relación $|(2m+1)\theta - (2\tilde{m}+1)\theta| \leq \theta$ que se deduce fácilmente de la anterior

$$|(2m+1)\theta - (2\tilde{m}+1)\theta| = |2m\theta - 2\tilde{m}\theta| = 2\theta|m - \tilde{m}| \leq 2\theta/2 = \theta. \tag{3.27}$$

Pero por la propia definición de $\tilde{m}$ se tiene que $(2\tilde{m}+1)\theta = \pi/2$, por lo tanto $|(2m+1)\theta - \pi/2| \leq \theta$. Con la relación entre ángulos complementarios se llega a que

$$\left|\sin\left((2m+1)\theta - \frac{\pi}{2}\right)\right| = |\cos((2m+1)\theta)|\,, \tag{3.28}$$

y utilizando la última propiedad

$$\left|\sin((2m+1)\theta) - \frac{\pi}{2})\right| \leq |\sin(\theta)|. \tag{3.29}$$

Juntando (3.28) y (3.29) se llega a la conclusión de que $|\cos((2m+1)\theta)| \leq |\sin(\theta)|$. Si ahora calculamos la probabilidad de hacer una medida errónea tras m iteraciones, es decir, la amplitud de todos los estados no marcados al cuadrado, se obtiene

$$(N-1)k_m^2 = \cos^2((2j+1)\theta) \leq \sin^2(\theta) = \frac{1}{N}. \tag{3.30}$$

Es decir la probabilidad de no medir el estado marcado tras m iteraciones esta acotada superiormente por $1/N$, que es un número pequeño.

Si nos fijamos otra vez en la definición de θ, podemos ver que $\theta = \arcsin\left(1/\sqrt{N}\right) \approx 1/\sqrt{N}$ y si sustituimos esto en $m = \lfloor \pi/4\theta \rfloor = \lfloor \frac{\pi}{4}\sqrt{N} \rfloor$[1]. Se ha obtenido la dependencia de tipo $O(\sqrt{N})$ que habíamos mencionado con anterioridad.

Ahora se puede mostrar todo este proceso de forma gráfica con una pequeña simulación del algoritmo. En está simulación se ha utilizado un estado con $2^8 = 256$ elementos y aunque el algoritmo está pensado para estados con más elementos, para visualizar el efecto serán suficientes 256.

Estas representaciones son las mostradas en la figura 3.4. En la primera de ellas 3.4a se parte de la figura 3.3c, que es el estado después de un paso del algoritmo. En las figuras 3.4a y 3.4b se puede ver como la amplitud del elemento marcado va aumentando con cada iteración hasta que llega al máximo en la figura 3.4c. Al tener un estado con $N = 256$ y tal y como se ha calculado en esta sección, el número óptimo de iteraciones será $\lfloor \frac{\pi}{4}\sqrt{N} \rfloor = \lfloor 4\pi \rfloor = 12$. Por lo tanto, hasta la iteración 12 la amplitud de los elementos no marcados va disminuyendo para que la amplitud del marcado aumente. En el siguiente paso, al haber alcanzado ya la amplitud máxima, el estado marcado debe disminuir, es lo que se muestra en la figura 3.4d.

Cuanto más nos acercamos a la máxima amplitud las ganancias de amplitud son cada vez menores por eso en las gráficas 3.4c y 3.4d apenas se distingue una iteración de la siguiente.

Para visualizar mejor la amplitud que tiene el estado marcado en cada iteración se muestra la figura 3.5. En esta representación se puede ver qué valor tiene la amplitud del estado marcado en cada iteración de una forma más clara que la representación 3.4. La linea roja de esta representación muestra la iteración en la que se alcanza la amplitud máxima, iteración 12. Al igual que en la figura 3.4d la amplitud del estado marcado cae a partir de la iteración 12.

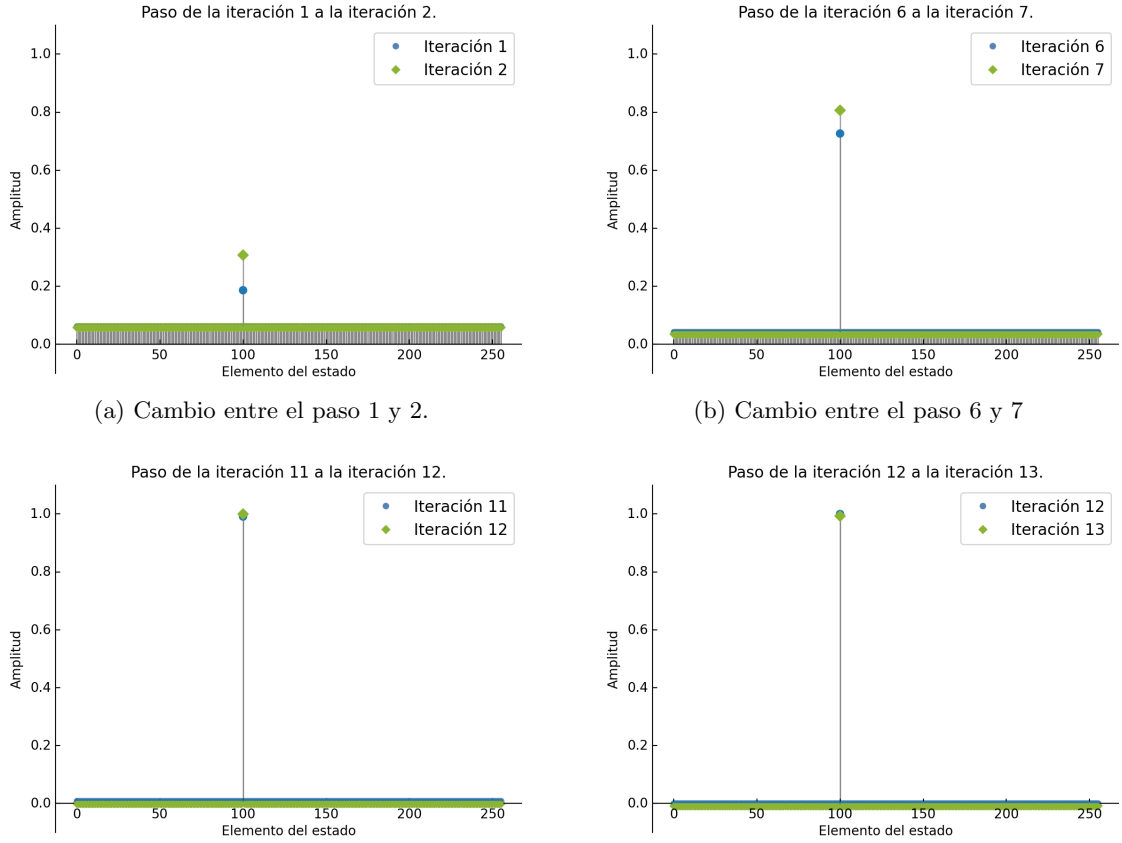
Podemos representar también la amplitud del estado marcado pero dejando actuar el algoritmo más pasos de los necesarios. De esta forma se debería observar que la amplitud evoluciona con una forma sinusoidal que debería coincidir con la dada en (3.26). En esta fórmula definíamos el ángulo que utilizábamos como $\sin^2\theta = 1/N$. Para N lo suficientemente grande podemos considerar que $\theta = \sin\theta = 1/\sqrt{N}$. En nuestro caso con $N = 256$ el error que se comete es de menos de 0,1 % por lo que apenas se debería apreciar la diferencia entre ambas.

Las amplitudes negativas en la figura 3.6 ocurren cuando el estado marcado tiene una fase de π con respecto de los no marcados, como ya se explicó que ocurría en la figura 3.3b. Esto se ve que también empieza a ocurrir en la figura 3.4d donde los estados no marcados empiezan a tener amplitudes negativas. En la figura 3.6 es difícil apreciar la diferencia entre los puntos que corresponden a la representación de la fórmula (3.26) y los que tienen que ver con nuestra simulación. Esto es una muestra de que la aproximación de $\theta = 1/\sqrt{N}$ es válida para $N = 256$.

3.3. Oráculo

En todo este capítulo se ha hablado de rotar la fase del estado marcado y gracias a eso conseguir que el resto de estados le cedan amplitud, pero se ha omitido deliberadamente el cómo detectar a qué estado le tenemos que rotar la fase. Esto es una cuestión primordial para el desarrollo de este algoritmo ya que lo que hace que las puertas (3.9) y (3.17) transfieran amplitud es precisamente la diferencia de fase que existe entre los estados. Si supiéramos de antemano a que estado hay que proporcionar una fase, todo el algoritmo carecería de sentido ya que se conocería ya el estado marcado y no haría falta buscarlo. Por ello, es necesario diseñar un sistema que sea capaz de reconocer si un estado es el que estamos buscando o

[1] Para el caso de que existan más elementos marcados se puede ver en [7] que el número de iteraciones óptimas será $m = \lfloor \frac{\pi}{4}\sqrt{N/S} \rfloor$, donde S es el número de elementos marcados. Esta fórmula muestra que cuantos más elementos marcados se tenga, menos iteraciones serán necesarias.



(a) Cambio entre el paso 1 y 2.

(b) Cambio entre el paso 6 y 7

(c) Cambio entre el paso 11 y 12. Máxima amplitud.

(d) Cambio entre el paso 12 y 13.

Figura 3.4: Variación de las amplitudes de los estados entre varias iteraciones.

no. Esto se hace a través de un oráculo o "black box", que es un concepto muy extendido en computación cuántica, utilizado en algoritmos como el de Shor [5] y el de Deutsch-Jozsa [8]. Es un concepto que parece un poco esotérico al principio pero es fácil de comprender.

El oráculo que tenemos que utilizar para este algoritmo es un oráculo que pueda distinguir si un estado es el marcado o no, algo similar ocurre en [5] donde tratamos de encontrar la factorización de un número cualquiera n. Para ello aumentamos la amplitud de los estados que corresponden a los números primos p y q tal que $p \cdot q = n$, esto sucede mediante un oráculo al que aportándole dos números puede decir si su multiplicación da como resultado el número n, pero no es capaz por sí solo de encontrar los dos números p y q que factorizan el número deseado. Podemos definir un oráculo como un sistema físico que es capaz de reconocer las soluciones. En el caso clásico del algoritmo de Grover también hay que utilizar un sistema que sea capaz de reconocer cuál de los N elementos corresponde con el que se está buscando. Un ejemplo práctico de esto es si se quiere encontrar la letra p de entre todas las letras del abecedario. El algoritmo sería el siguiente: se introduce por teclado la letra que queremos encontrar, p en este caso, el ordenador va comparando cada una de las letras del abecedario con la introducida por teclado (esto correspondería con la llamada al oráculo) en el momento en el que coincida la búsqueda ha finalizado. En este algoritmo clásico se llama al oráculo $O(N)$ veces para comprobar si el elemento coincide, sin embargo con el algoritmo clásico esta llamada al oráculo se reduce hasta $O(\sqrt{N})$ veces.

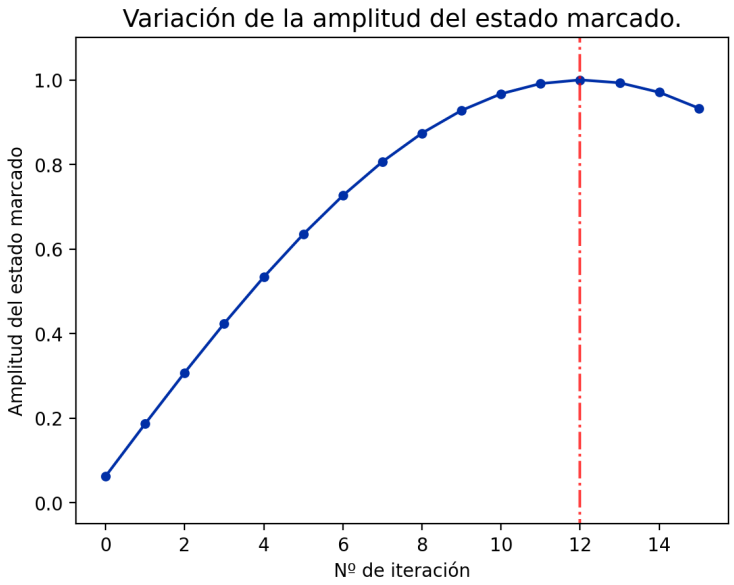


Figura 3.5: Evolución de la amplitud del estado marcado.

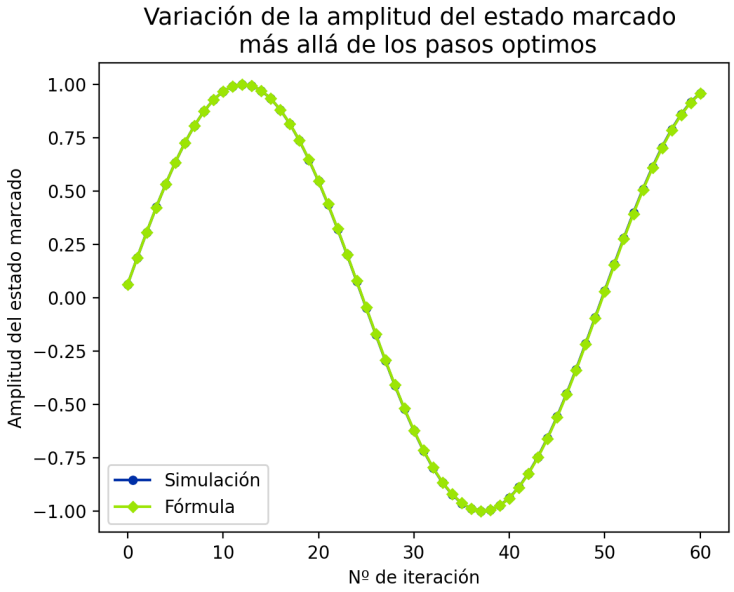


Figura 3.6: Comparación de la evolución completa de la amplitud del estado marcado entre la simulación y la ecuación (3.26).

Capítulo 4

Proyecciones

Habiéndose tratado ya el algoritmo de Grover con la rotación y la difusión en este capítulo se trata de analizar brevemente otro punto de vista que da lugar a los mismos resultados.

4.1. Descripción geométrica del algoritmo de Grover

En la sección 3.2 se trata el algoritmo con la matriz D (3.17) y la matriz R tal y como viene expuesto en [4]. Otra forma de definir estas operaciones es utilizando proyectores de estados que ya se conocen o que están presentes en el problema, como se hace en el capítulo 4 del libro de Mermin [9]. Cuando en la sección 3.2 se definía la matriz D como $D = -\mathbb{1} + 2P$ ya se estaban utilizando los proyectores pero no se especificaba de qué estado eran proyectores. Para seguir la misma notación que se utiliza en el libro se va a llamar a la difusión W y a la rotación V. Para definir estos operadores en forma de proyectores será necesario primero definir los estados. El primero será la superposición uniforme de todos los estados (3.19) que llamaremos $|\phi_0\rangle$. El siguiente estado será una superposición uniforme de los elementos marcados

$$|\psi\rangle = \frac{1}{\sqrt{S}} \sum_{x \in S} |x\rangle \,, \tag{4.1}$$

donde S juega, a la vez y sin posible, confusión el papel de número de elementos marcados y conjunto de elementos marcados. Utilizando estos estados se pueden definir los operadores W y V (antes llamados D y R respectivamente):

$$W = -\mathbb{1} + 2\,|\phi_0\rangle\,\langle\phi_0| \qquad \text{y} \qquad V = \mathbb{1} - 2\,|\psi\rangle\,\langle\psi| \,. \tag{4.2}$$

Para ver la acción de estos operadores se va a suponer que solo existe un único elemento marcado, por lo tanto la expresión (4.1) queda como $|\psi\rangle = |a\rangle$ siguiendo la notación dada por [9]. Se va a empezar viendo la acción de W sobre $|\phi_0\rangle$ y $|a\rangle$:

$$\begin{aligned} W\,|\phi_0\rangle &= -\,|\phi_0\rangle + 2\,|\phi_0\rangle\,\langle\phi_0|\phi_0\rangle = |\phi_0\rangle \,, \\ W\,|a\rangle &= -\,|a\rangle + 2\,|\phi_0\rangle\,\langle\phi_0|a\rangle = -\,|a\rangle + \tfrac{2}{\sqrt{N}}\,|\phi_0\rangle \,. \end{aligned} \tag{4.3}$$

Podemos ver que el operador W deja invariante a $|\phi_0\rangle$, lo que es fácil de entender si se piensa en W como el operador inversión sobre la media actuando sobre una superposición uniforme. Esto se ha calculado

teniendo en cuenta que $|a\rangle$ es una de las componentes de la superposición uniforme $|\phi_0\rangle$. Ahora toca calcular la acción de V:

$$V|\phi_0\rangle = |\phi_0\rangle - 2|a\rangle\langle a|\phi_0\rangle = |\phi_0\rangle - \tfrac{2}{\sqrt{N}}|a\rangle, \qquad V|a\rangle = |a\rangle - 2|a\rangle\langle a|a\rangle = -|a\rangle. \tag{4.4}$$

Con lo que se puede comprobar que efectivamente el operador V corresponde con la rotación de la sección 3.2 ya que rota el estado marcado dejando al resto intactos.

Partiendo de las expresiones (4.3) y (4.4) se puede calcular cual es la acción conjunta de WV sobre el estado inicial $|\phi_0\rangle$.

$$WV|\phi_0\rangle = W\left(|\phi_0\rangle - \frac{2}{\sqrt{N}}|a\rangle\right) = |\phi_0\rangle - \frac{2}{\sqrt{N}}\left(-|a\rangle + \frac{2}{\sqrt{N}}|\phi_0\rangle\right) = \left(1-\frac{4}{N}\right)|\phi_0\rangle + \frac{2}{\sqrt{N}}|a\rangle. \tag{4.5}$$

Teniendo en cuenta que el estado $|a\rangle$ está incluido en estado $|\phi_0\rangle$ se puede calcular cuál será la nueva amplitud del estado marcado y compararla con la que obtendríamos utilizando la expresión (3.25) para calcular K_1 utilizando $K_0 = k_0 = 1/\sqrt{N}$:

$$\left(1-\tfrac{4}{N}\right)\tfrac{1}{\sqrt{N}} + \tfrac{2}{\sqrt{N}} \stackrel{?}{=} \tfrac{N-2}{N\sqrt{N}} + \tfrac{2(N-1)}{N\sqrt{N}} \qquad \tfrac{3N-4}{N\sqrt{N}} = \tfrac{3N-4}{N\sqrt{N}}. \tag{4.6}$$

Con lo que queda demostrado que es equivalente esta interpretación del algoritmo de Grover con la dada en la sección 3.2.

En (4.3) y (4.4) se observa que la acción de W y V sobre $|\phi_0\rangle$ y $|a\rangle$ da como resultado combinaciones lineales de estos dos estados. Por lo tanto se puede definir una base de un espacio con los estados $|a\rangle$ y $|a_\perp\rangle$ siendo $|a_\perp\rangle$ todos los estados ortogonales a $|a\rangle$ que están contenidos en $|\phi_0\rangle$. Al ser (4.3) y (4.4) combinaciones lineales de $|\phi_0\rangle$ y $|a\rangle$ todos los estados generados por la acción de W y V sobre elementos de este espacio pertenecen al espacio $\{|a\rangle, |a_\perp\rangle\}$. Si nos imaginamos este espacio como un plano bidimensional se podrá visualizar el algoritmo de Grover de forma más sencilla.

En la figura 4.1a se puede ver una representación del espacio bidimensional generado por $|a\rangle$ y $|a_\perp\rangle$ donde el estado $|\phi_0\rangle$ es una combinación lineal de estos dos vectores.

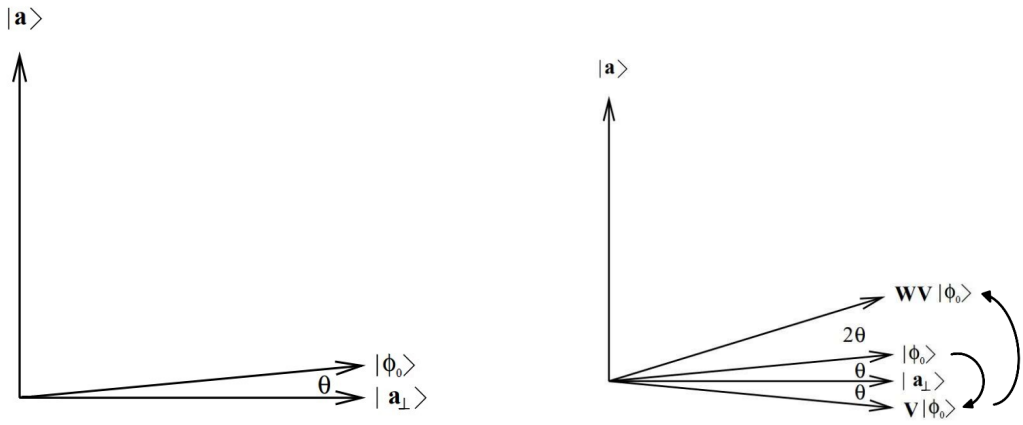
El estado $|\phi_0\rangle$ es muy próximo a $|a_\perp\rangle$ ya que la proyección sobre $|a\rangle$ ($\langle\phi_0|a\rangle = 1/\sqrt{N}$)[1] es pequeña cuando N es grande. Si se utiliza esta representación se puede observar que la acción de V sobre un elemento de este espacio es devolver la reflexión de ese estado sobre el vector $|a_\perp\rangle$, es por eso que $V|a\rangle = -|a\rangle$. También es sencillo ver que la acción de W será la reflexión sobre la linea marcada por el estado $|\phi_0\rangle$.

Como se puede ver en la figura 4.1b el resultado de $WV|\phi_0\rangle = |\phi_1\rangle$ es un vector en el que su componente $|a\rangle$ tiene más peso. Si se vuelve a aplicar WV sobre $|\phi_1\rangle$ se obtendrá un estado que será más próximo a $|a\rangle$, por lo tanto, con las suficientes aplicaciones nos podremos acercar al estado $|a\rangle$ tanto como queramos, al igual que pasaba en la sección 3,2.

El valor de $\langle\phi_0|a\rangle$ está representado en la figura 4.1a como el $\sin\theta$. Se sabe que este valor es $1/\sqrt{N}$, que es pequeño, por lo que se puede aproximar $\theta = 1/\sqrt{N}$. Utilizando este ángulo y sabiendo que en la i-ésima aplicación de WV el estado $|\phi_i\rangle$ formará un ángulo de $(2i+1)\theta$ con el estado $|a_\perp\rangle$ podemos calcular que ángulo formarán después de $\lfloor\frac{\pi}{4}\sqrt{N}\rfloor$ aplicaciones, que es el valor obtenido en la sección 3.2.

$$\left(2\frac{\pi}{4}\sqrt{N}+1\right)\frac{1}{\sqrt{N}} = \frac{\pi}{2} + \frac{1}{\sqrt{N}}. \tag{4.7}$$

[1] Aquí se está considerando que solo hay un elemento marcado. En el caso de que hubiera más de un elemento marcado habría que tener en cuenta $\langle\phi_0|\psi\rangle = S/\sqrt{N}$.



(a) Representación de $|\phi_0\rangle$, $|a\rangle$ y $|a_\perp\rangle$.

(b) Representación de la acción de W y V.

Figura 4.1: Representación del espacio bidimensional $\{|a\rangle, |a_\perp\rangle\}$. Imagen obtenida de [9]

Teniendo en cuenta que $\sqrt{N}$ es grande este ángulo es $\pi/2$ confirmando así que el estado $|\phi_0\rangle$ después de $\lfloor \frac{\pi}{4}\sqrt{N} \rfloor$ aplicaciones de WV será prácticamente el estado $|a\rangle$.

En esta sección se ha visto una forma alternativa de llegar al mismo algoritmo que llega Grover en [4]. La idea de este trabajo es, partiendo de la idea planteada en esta sección, llegar a otra forma del algoritmo de Grover utilizando medidas cuánticas. Para ello vamos a intentar utilizar a nuestro favor el efecto Zenón cuántico, el cual vamos a explicar en la siguiente sección.

4.2. Efecto Zenón cuántico

Otro ejemplo del uso del efecto Zenón cuántico, que es más útil para el propósito de este trabajo, es el mostrado en [10]. En este artículo diseñan un sistema para hacer medidas sin interacción entre un fotón y el objeto, utilizando óptica cuántica. Y en un punto del artículo plantean la idea de evitar que el estado de polarización de un fotón cambie utilizando el efecto Zenón cuántico. El esquema se muestra en la figura 4.2 y es el siguiente: un fotón con polarización horizontal pasa por una serie láminas que rotan un ángulo θ pequeño la polarización de este fotón, haciendo que acabe con polarización vertical. Al final del montaje se encuentra un polarizador horizontal y detrás un sensor. Tal y como está montado el sistema el fotón inicial con polarización horizontal acaba con polarización vertical y al encontrarse con el polarizador horizontal no puede atravesarlo y el sensor no detecta nada. El efecto Zenón cuántico puede aprovecharse si después de cada lámina rotora colocamos un polarizador horizontal, de tal forma que si el estado después de atravesar la lamina rotora no difiere mucho de polarización horizontal, al estar rotado solo un ángulo θ pequeño, con mucha probabilidad va a atravesar el polarizador horizontal y volver al estado de polarización inicial. Si esto se repite después de cada rotación de la polarización se llegará al final con una polarización horizontal, haciendo así que el sensor detecte el fotón, suponiendo que el fotón no haya sido absorbido por uno de los polarizadores horizontales.

Utilizando este fenómeno se ha conseguido "congelar" la evolución del estado, al igual que en el experimento expuesto en [11]. El efecto que es interesante para este trabajo es justo el contrario: si se ha podido evitar la evolución de un estado utilizando medidas sucesivas, se podría también forzar la evolución del sistema utilizando medidas. Si en el esquema inferior de la figura 4.2 se retiran las láminas polarizadoras (*polarization rotators*) y los polarizadores horizontales se van rotando un ángulo θ con

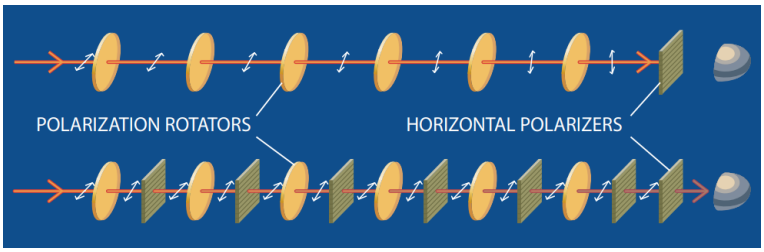


Figura 4.2: Esquema óptico. Imagen obtenida de [10].

respecto al anterior sucesivamente, lo que ocurrirá será que el fotón atravesará el primer polarizador con una probabilidad $\cos^2\theta$. Una vez atravesado ese polarizador estará en un estado de polarización que forme un ángulo de θ radianes con la polarización horizontal. Si se consigue atravesar todos los polarizadores, lo que ocurrirá con una probabilidad $(\cos^2\theta)^n$ donde n es el número de polarizadores, es que el fotón saldrá del sistema con una polarización vertical. Teniendo esto en cuenta se puede ver que haciendo el ángulo tan pequeño como se quiera y por lo tanto elevando el número de pasos, se podría sobrevivir con una probabilidad tan cercana a 1 como se quiera. El ángulo θ y el número de pasos están relacionados en este ejemplo ya que si se quiere llegar con n rotaciones de ángulo θ a $\pi/2$, $\theta = \pi/2n$.

Utilizando este fenómeno podemos forzar que un sistema evolucione hasta un estado deseado. Siempre que se utilicen medidas para hacer evolucionar un sistema habrá que tener en cuenta que no siempre obtendremos el sistema deseado, ya que la medida que hacemos no puede tener probabilidad 1 de obtener el resultado que queremos. En caso de que la probabilidad de obtener el resultado que queremos sea 1 el sistema no evolucionará y estaremos en el caso simple de efecto Zenón cuántico. Haciendo uso de esta propiedad tal vez sea posible encontrar un algoritmo de Grover con medidas.

4.3. Algoritmo con proyectores

Se ha visto en la sección anterior que es posible hacer que un sistema evolucione haciendo medidas sucesivas. Para construir un algoritmo de Grover se necesita que en esta evolución se transfiera amplitud desde unos estados hasta otros. Esto es posible hacerlo utilizando proyectores como se va a mostrar en esta sección por lo que creemos que podría ser posible hacerlo con medidas.

En la sección 4.1 se empieza con el estado $|\phi_0\rangle$, que es la superposición uniforme de todos los estados, y para llegar al siguiente estado se aplican los operadores W y V.

$$|\phi_1\rangle = WV\,|\phi_0\rangle\,. \tag{4.8}$$

Siempre se puede construir el proyector de un estado multiplicando el ket por el bra.

$$P_1 = |\phi_1\rangle\langle\phi_1| = WV\,|\phi_0\rangle\,(WV\,|\phi_0\rangle)^\dagger = WV\,|\phi_0\rangle\langle\phi_0|\,V^\dagger W^\dagger. \tag{4.9}$$

Teniendo ya el proyector construido se sabe que este proyector tiene que corresponder a un observable y por lo tanto a una medida. Esto se puede afirmar ya que un observable A se puede construir como suma de proyectores ponderada.

$$A = \sum_n a_n P_n. \tag{4.10}$$

Donde a_n corresponde al autovalor del observable A para el subespacio $\mathscr{E}_n$ del que P_n es proyector. Por lo tanto teniendo un proyector P_n se puede construir una medida con dos posibles resultados, uno

correspondiente a P_n y el otro a $\mathbb{1}-P_n$, ya que $\mathbb{1}-P_n$ es también un proyector. Sabiendo que se corresponde a un posible resultado de una medida se puede realizar dicha medida y si se obtiene el resultado esperado el estado después de la medida será

$$\frac{P_1 |\phi_0\rangle}{\sqrt{\langle\phi_0| P_1 |\phi_0\rangle}} = |\phi_1\rangle , \tag{4.11}$$

como cabría esperar. El estado $|\phi_1\rangle$ equivale al estado $WV |\phi_0\rangle$ que está representado en la figura 4.1b por lo tanto al aplicar el proyector lo que se está haciendo es proyectar el estado $|\phi_0\rangle$ sobre el eje $WV |\phi_0\rangle$. Al hacer esta operación existe la posibilidad de fallar y no conseguir el resultado deseado en la medida. La probabilidad de que sí se obtenga el resultado deseado es

$$\langle\phi_0| P_1 |\phi_0\rangle = \left(1 - \frac{2}{N}\right)^2 \tag{4.12}$$

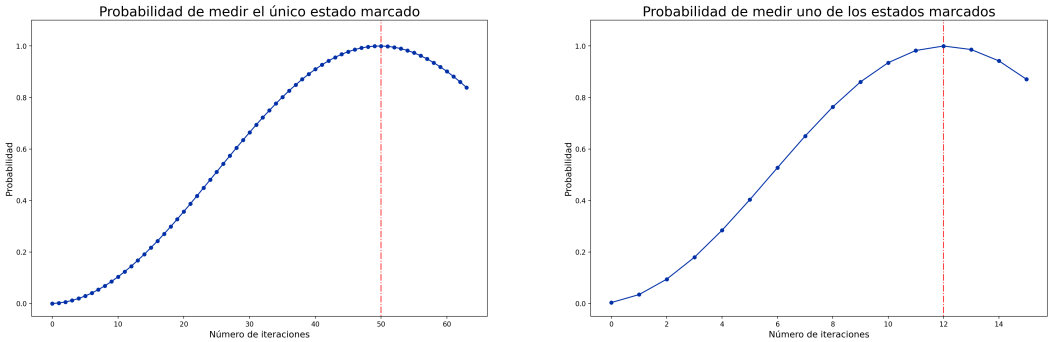
para el caso de que exista un solo elemento marcado. El siguiente estado en la sección 4.1 se conseguía volviendo a aplicar los operadores W y V al estado $|\phi_1\rangle$, en este caso se procede definiendo un nuevo proyector

$$P_2 = |\phi_2\rangle\langle\phi_2| = WV |\phi_1\rangle\langle\phi_1| V^\dagger W^\dagger = (WV)^2 |\phi_0\rangle\langle\phi_0| (V^\dagger W^\dagger)^2, \tag{4.13}$$

y volviendo a aplicar el postulado 5 con el proyector P_2 y el estado $|\phi_1\rangle$.

Siguiendo este esquema se consigue el mismo efecto que el algoritmo de Grover usual, pero teniendo en cuenta que cabe la posibilidad de que una medida arroje un resultado erróneo. Se podría considerar una implementación óptica de este algoritmo similar a la representada en la figura 4.2. Si se considera el estado inicial $|\phi_0\rangle$ como un estado con una polarización prácticamente horizontal, colocando un polarizador a 2θ con respecto del estado inicial se conseguiría el estado $|\phi_1\rangle$. Si se van rotando los polarizadores sucesivos un ángulo de 2θ con respecto de los anteriores, al final se conseguiría un estado muy próximo a $|a\rangle$.

Al igual que en el algoritmo usual el último paso consiste en una medida en la que con mucha probabilidad se encuentra el o uno de los estados marcados. Se puede tratar de hacer una representación equivalente a la figura 3.5 pero esta vez representando la probabilidad de medir el marcado teniendo para $N = 4096$. La linea roja en ambas representaciones marca cuándo se consigue la máxima probabilidad teóricamente. Para el caso de un solo estado marcado esto viene dado por $\lfloor \frac{\pi}{4}\sqrt{4096} \rfloor = 50$ y para el caso de varios estado marcados por $\lfloor \frac{\pi}{4}\sqrt{N/S} \rfloor = \lfloor \frac{\pi}{4}\sqrt{4096/12} \rfloor = 12$, como se mencionaba en el pie de página 1.



(a) $N = 4096$ y $S = 1$. (b) $N = 4096$ y $S = 16$.

Figura 4.3: Probabilidad de medir el estado marcado con proyectores para varios casos.

Esta representación tiene trampa ya que nos muestra la probabilidad de medir el estado marcado suponiendo que se haya llegado hasta esa iteración. Como ya se mencionó no se tiene la certeza absoluta

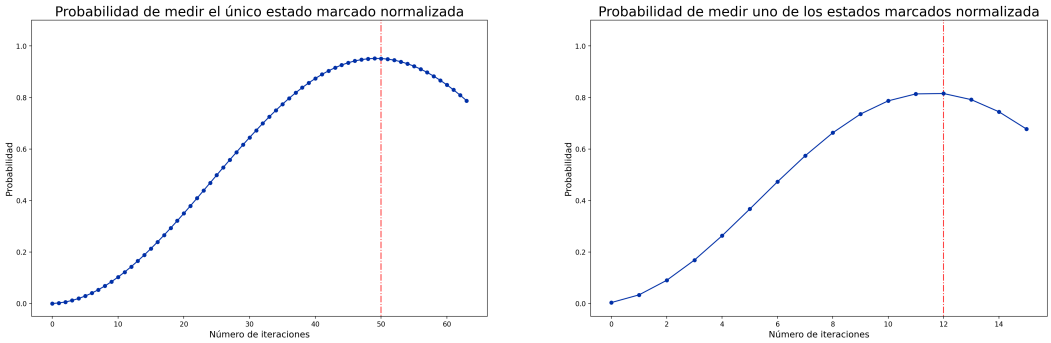
de obtener siempre el resultado ideal, por lo que las gráficas 4.3 tienen que ser normalizadas por la probabilidad de sobrevivir hasta esa iteración. Esta probabilidad se calcula teniendo en cuenta que la probabilidad de la medida satisfactoria en la iteración i es $\langle\phi_i| P_{i+1} |\phi_i\rangle$. Por lo tanto la probabilidad de sobrevivir hasta la emésima iteración será

$$P(sob)_m = \prod_{i=0}^{m} \langle\phi_i| P_{i+1} |\phi_i\rangle \,. \tag{4.14}$$

Habiendo definido así la probabilidad de sobrevivir, se puede definir una nueva probabilidad llamada de éxito que nos de cuenta de la probabilidad de que el algoritmo aporte un resultado satisfactorio. Esta probabilidad será la probabilidad de sobrevivir hasta cierta iteración por la probabilidad de medir el estado marcado en esa iteración.

$$P(ex)_m = \langle\phi_m|\psi\rangle \langle\psi|\phi_m\rangle \prod_{i=0}^{m} \langle\phi_i| P_{i+1} |\phi_i\rangle \,, \tag{4.15}$$

dado que la probabilidad de medir el estado marcado es $\langle\phi_m|P_\psi|\phi_m\rangle$ [2]. Si se representan los valores de la probabilidad (4.15) se obtendrán las gráficas 4.3 pero normalizadas.



(a) $P(ex)$ con $N = 4096$ y $S = 1$.

(b) $P(ex)$ con $N = 4096$ y $S = 16$.

Figura 4.4: Probabilidad de medir el estado marcado con proyectores por la probabilidad de sobrevivir hasta esa iteración para varios casos.

Se ve que la gráfica 4.4a no difiere tanto de la representación 4.3a ya que con un solo estado marcado la probabilidad de sobrevivir es grande debido a que el ángulo θ es pequeño. Sin embargo hay más diferencia entre las representaciones 4.3b y 4.4b ya que en este caso el ángulo formado entre $|a_\perp\rangle$ y $|\phi_0\rangle$ es mayor y por lo tanto la probabilidad de hacer la rotación desde $|\phi_0\rangle$ hasta $|\phi_1\rangle$ es menor.

[2] Recordar que el estado $|\psi\rangle$ es la superposición uniforme de todos los estados marcados (4.1)

Capítulo 5

Evolución a partir de medidas cuánticas

Una vez expuesto que debería ser posible hacer el algoritmo de Grover utilizando medidas, se va a tratar de encontrar un oráculo que combinado con una serie de medidas consiga este efecto. Como ya se comentó en la sección 3.3 el oráculo es un sistema físico que es capaz de distinguir los elementos marcados. En un algoritmo clásico también es necesario el uso de un oráculo, ya que gracias a él se van comprobando 1 por 1 todas las posibles. En este trabajo no se va a tratar de diseñar un oráculo físicamente, simplemente se va a discutir cuál debería ser el funcionamiento del mismo. El objetivo es construir un sistema que sea capaz de, al hacer una medida, redistribuir la amplitud desde los estados no marcados hasta los marcados de tal forma que después de una serie de pasos se haga otra medida y se obtenga con mucha probabilidad uno de los estados marcados.

Al contrario que en el resto de secciones en esta se van a desarrollar todos los cálculos teniendo en cuenta que puede existir más de una solución, para ello se va a hacer un repaso de la notación que se utiliza.

- n hace referencia al número de qubits que estemos tratando.
- $N = 2^n$ es el número total de posibles estados.
- S hace referencia al conjunto de todos los estados que forman parte de la solución pero también al número de estados que forman parte de la solución, es decir el cardinal del conjunto S.
- NS es equivalente a S pero con los estados que no pertenecen a la solución.
- $|\psi\rangle\,|\phi\rangle$ es una abreviatura de $|\psi\rangle \otimes |\phi\rangle$.
- Todos los kets que se escriban en esta sección hacen referencia a estados de n qubits. Por lo tanto el ket $|\mathbf{0}\rangle$ hace referencia a $|00\cdots0\rangle$. En caso de que haya algún ket que no se refiera al estado de n qubits se indicará poniendo explícitamente como subíndice el número de qubits al que hace referencia.
- Los kets $|\phi_i\rangle$ hacen referencia a los estados antes de aplicar el oráculo y el subíndice i hace referencia al número de medidas que se han realizado.
- Los kets $|\Psi_i\rangle$ son equivalentes a los kets $|\phi_i\rangle$ pero hacen referencia a los estados después de aplicar el oráculo.

5.1. Rotación de los estados no marcados

Nuestro primer oráculo parte del mismo punto del que se ha comenzado en las anteriores secciones, una superposición uniforme de todos los estados. Por lo tanto considerando n qubits

$$|\phi_0\rangle = H^{\otimes n}\,|00\cdots0\rangle = \frac{1}{\sqrt{N}}\sum_{x=0}^{N}|x\rangle\,. \tag{5.1}$$

Dentro de esta superposición existirán estados que son parte de la solución ($x \in S$) y estados que no lo son ($x \notin S$). Esto se puede expresar

$$|\phi_0\rangle = \frac{1}{\sqrt{N}}\sum_{x\in S}|x\rangle + \frac{1}{\sqrt{N}}\sum_{x\notin S}|x\rangle\,. \tag{5.2}$$

El cambio desde la ecuación (5.1) hasta la ecuación (5.2) es meramente un cambio visual. Todos los estados siguen teniendo la misma amplitud pero los expresamos así para ver más claramente cuáles son los que nos interesan y cuáles no. El siguiente paso es la llamada al oráculo. En este paso se añade un segundo registro iniciado en $|00\cdots0\rangle = |\mathbf{0}\rangle$ y a los elementos que no nos interesan se les aplica una rotación.

$$|\Psi_0\rangle = \sum_{x\in S}\alpha_0\,|x\rangle\,|\mathbf{0}\rangle + \sum_{\substack{x\notin S\\ x\neq 0}}\alpha_0\cos\theta\,|x\rangle\,|\mathbf{0}\rangle + \sum_{\substack{x\notin S\\ x\neq 0}}\alpha_0\sin\theta\,|\mathbf{0}\rangle\,|x\rangle + \alpha_0\,|\mathbf{0}\rangle\,|\mathbf{0}\rangle\,, \tag{5.3}$$

donde se ha denotado la amplitud inicial $1/\sqrt{N}$ como α_0. En esta expresión se ha tratado el elemento $|\mathbf{0}\rangle\,|\mathbf{0}\rangle$ por separado para asegurar que el estado (5.3) esté normalizado. Esto es necesario ya que, al añadir un segundo registro que está iniciado en el estado $|\mathbf{0}\rangle$, el elemento del estado total correspondiente a $|\mathbf{0}\rangle\,|\mathbf{0}\rangle$ aparecerá dos veces, una en el sumatorio con la parte $\cos\theta$ y otra en el tercer sumatorio con la parte $\sin\theta$. Al calcular la norma de este estado $|\,\langle\phi_0|\phi_0\rangle\,|^2$ aparecerían términos del tipo $\cos\theta\sin\theta$ harían que el estado no estuviera normalizado. Esto ocurrirá suponiendo que $|\mathbf{0}\rangle \notin S$, pero en caso contrario los resultados no varían por lo que se va a desarrollar este algoritmo considerando el último elemento por separado.

Una vez que se ha aplicado el oráculo por primera vez tendremos que hacer nuestra primera medida. La medida que puede resultar beneficiosa es una medida del segundo registro o segundo qubit. Se considera una medida exitosa si se mide el estado $|\mathbf{0}\rangle$ en el segundo registro, ya que si se obtiene un estado $|x\rangle\,|x \notin S$ el estado quedaría perfectamente definido y no se podría operar más. La probabilidad de medir $|\mathbf{0}\rangle$ en el segundo registro será 1 menos la probabilidad de medir algo distinto de cero.

$$1 - \frac{1}{N}(NS-1)\sin^2\theta = |\alpha_1|^2. \tag{5.4}$$

Llamamos α_1 a la amplitud de probabilidad de medir el estado distinto de cero. Si se ha conseguido un resultado satisfactorio en esta medida el siguiente paso es normalizar el estado en virtud del postulado 5. El resultante será

$$|\phi_1\rangle = \frac{1}{\alpha_1}\left(\sum_{x\in S}\alpha_0\,|x\rangle\,|\mathbf{0}\rangle + \sum_{\substack{x\notin S\\ x\neq 0}}\alpha_0\cos\theta\,|x\rangle\,|\mathbf{0}\rangle + \alpha_0\,|\mathbf{0}\rangle\,|\mathbf{0}\rangle\right), \tag{5.5}$$

que está normalizado. Ahora se vuelve a repetir el procedimiento que nos llevó a (5.3), una llamada al oráculo, pero teniendo en cuenta que el estado (5.5) es ahora nuestro estado (5.2).

$$|\Psi_1\rangle = \frac{1}{\alpha_1}\left(\alpha_0 \sum_{x\in S}|x\rangle|\mathbf{0}\rangle + \alpha_0\cos^2\theta\sum_{\substack{x\notin S\\ x\neq 0}}|x\rangle|\mathbf{0}\rangle + \alpha_0\sin\theta\cos\theta\sum_{\substack{x\notin S\\ x\neq 0}}|\mathbf{0}\rangle|x\rangle + \alpha_0|\mathbf{0}\rangle|\mathbf{0}\rangle\right). \tag{5.6}$$

Si otra vez repetimos la medida del segundo registro la probabilidad de obtener algo distinto de cero es

$$|\alpha_2|^2 = 1 - \left|\frac{\alpha_0}{\alpha_1}\right|^2 (NS-1)\sin^2\theta\cos^2\theta. \tag{5.7}$$

Ahora se vuelve a normalizar obteniendo así

$$|\phi_2\rangle = \frac{1}{\alpha_2\alpha_1}\left(\alpha_0 \sum_{x\in S}|x\rangle|\mathbf{0}\rangle + \alpha_0\cos^2\theta\sum_{\substack{x\notin S\\ x\neq 0}}|x\rangle|\mathbf{0}\rangle + \alpha_0|\mathbf{0}\rangle|\mathbf{0}\rangle\right). \tag{5.8}$$

Todo esto se va repitiendo siguiendo el mismo proceso. Lo que nos interesa al final es medir el primer registro y encontrar uno de los estados marcados. Es importante darse cuenta de que solo vamos a obtener uno de los estados pertenecientes a la solución, esto es común a cualquier proceso de Grover. La probabilidad de encontrar uno de los estados marcados va cambiando con cada iteración. En (5.5) la probabilidad es de $S\left|\frac{\alpha_0}{\alpha_1}\right|^2$ y en (5.8) $S\left|\frac{\alpha_0}{\alpha_1\alpha_2}\right|^2$. Podemos comprobar que la probabilidad depende del valor de α_i, que es la amplitud de probabilidad de medir $|\mathbf{0}\rangle$ en el segundo registro en la i-ésima medida. Nos será útil encontrar una fórmula para calcular el valor de α_i para cualquier valor de i. Para ello vemos como sería el estado antes de la i-ésima medida.

$$|\phi_{i-1}\rangle = \frac{1}{\prod_{j=1}^{i-1}\alpha_j}\left(\alpha_0 \sum_{x\in S}|x\rangle|\mathbf{0}\rangle + \alpha_0\cos^{i-1}\theta\sum_{\substack{x\notin S\\ x\neq 0}}|x\rangle|\mathbf{0}\rangle + \alpha_0|\mathbf{0}\rangle|\mathbf{0}\rangle\right). \tag{5.9}$$

Y después de aplicar el oráculo

$$|\Psi_{i-1}\rangle = \frac{1}{\prod_{j=1}^{i-1}\alpha_j}\left(\alpha_0 \sum_{x\in S}|x\rangle|\mathbf{0}\rangle + \alpha_0\cos^{i}\theta\sum_{\substack{x\notin S\\ x\neq 0}}|x\rangle|\mathbf{0}\rangle + \alpha_0\sin\theta\cos^{i-1}\theta\sum_{\substack{x\notin S\\ x\neq 0}}|\mathbf{0}\rangle|x\rangle + \alpha_0|\mathbf{0}\rangle|\mathbf{0}\rangle\right). \tag{5.10}$$

De esta expresión es fácil sacar cuál será el valor de α_i.

$$|\alpha_i|^2 = 1 - \frac{|\alpha_0|^2}{|\prod_{j=1}^{i-1}\alpha_j|^2}(NS-1)\sin^2\theta\cos^{2i-2}\theta. \tag{5.11}$$

Claramente esta fórmula depende del ángulo θ que escojamos por lo que se obtendrán distintos resultados dependiendo del ángulo.

El objetivo es medir uno de los estados marcados, para ello será necesario una medida del primer registro. La probabilidad de medir un estado marcado en el estado (5.3) es la misma que en (5.1) y (5.2) ya que no se ha actuado sobre la parte $x \in S$.

$$P(mar)_0 = S|\alpha_0|^2 = \frac{S}{N}. \tag{5.12}$$

Esto representa la probabilidad de que al escoger un elemento de la superposición uniforme se obtenga uno del marcado directamente. El objetivo de este algoritmo es que esta probabilidad de medir un elemento marcado vaya aumentando, de tal forma que cada vez sea más probable medir un elemento marcado. En el estado (5.5) la probabilidad de medir un elemento marcado será

$$P(mar)_1 = S\left|\frac{\alpha_0}{\alpha_1}\right|^2 = \frac{S}{N|\alpha_1|^2}. \tag{5.13}$$

Como la amplitud $\alpha_1 \leq 1$ podemos asegurar que $P(mar)_1 \geq P(mar)_0$. El caso de que sea igual sucederá cuando $\alpha_1 = 1$, es decir $\sin\theta = 0$.

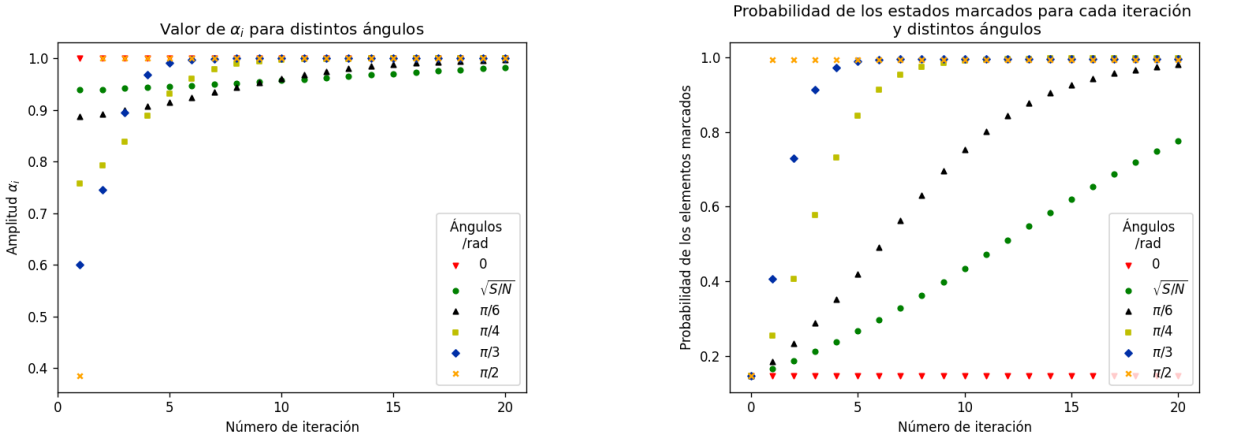
Entre los estados (5.5) y (5.6) la amplitud de los elementos marcados no cambia por lo que será indiferente en cuanto a probabilidades medir el primer registro antes o después, la única diferencia será en cuántas veces se aplica el oráculo. Si se calcula la probabilidad $P(mar)_2$ de medir el marcado después de la segunda medida se obtiene:

$$P(mar)_2 = S\left|\frac{\alpha_0}{\alpha_1\alpha_2}\right|^2 = \frac{S}{N|\alpha_1\alpha_2|^2}. \tag{5.14}$$

Siguiendo el patrón se puede ver que la probabilidad de medir el marcado en $|\phi_i\rangle$ o $|\Psi_i\rangle$ es

$$P(mar)_i = \frac{S}{N|\prod_{j=1}^{i}\alpha_j|^2}. \tag{5.15}$$

Podemos representar cómo será esta probabilidad para distintos ángulos y comprobarlo con los valores de α_i para los mismos ángulos.



(a) Valor de los α_i

(b) Representación de $P(mar)_i$ para distintos ángulos

Figura 5.1: Valores de las expresiones (5.11) y (5.15) para distintos ángulos con valores $N = 1024$, $S = 150$.

En la figura 5.1 se puede ver cómo varían los valores de las expresiones (5.11) y (5.15) para $\theta = 0, \pi/6, \pi/4, \pi/3$ y $\pi/2$. También se ha representado el valor de $\theta = \sqrt{S/N}$ que es el valor que debería tomar el ángulo de (3.26) para el caso en el que haya más de una posible solución [7]. Se observa en la figura 5.1a que para $\theta = 0$ la probabilidad de medir $|\mathbf{0}\rangle$ en el segundo registro es siempre 1 ya que $\sin(0) = 0$. Al ser todos los $\alpha_i = 1$ la probabilidad de medir un elemento marcado no varia en ningún momento y es siempre igual a $S/N = 150/1024 = 0{,}1465$, que es el valor que se representa en 5.1b.

Para los valores de $\theta = \pi/6, \pi/4, \pi/3$ y $\sqrt{S/N}$ ocurre algo similar en todos. Los valores de α_i adquieren rápidamente un valor entre 0.9 y 1. En el caso de $\sqrt{S/N}$ y $\pi/6$ se puede ver en la figura 5.1a que todos los α_i tienen valores altos, por lo que tiene sentido que en la figura 5.1b sean las gráficas que más lento crecen, ya que el denominador de la expresión (5.15) será más grande que en otros casos. Ocurre justo lo contrario para $\theta = \pi/4$ y $\theta = \pi/3$.

En el caso de $\theta = \pi/2$ el primer valor que se representa en 5.1a es el correspondiente con (5.4) cuando $\sin\theta = 1$. Para calcular el segundo punto ya hay que tener en cuenta el $\cos\theta = 0$ (5.7), es por eso que el valor de α_2 en adelante es 1. A partir de esa iteración los valores corresponden con el caso $\theta = 0$. También se puede observan en la figura 5.1b que para la iteración 0 todas las probabilidades valen lo mismo, que es el valor expresado en (5.12).

Como se hizo en la sección 4.3 hay que tener en cuenta la probabilidad de sobrevivir hasta una cierta iteración. Por lo tanto el equivalente a la expresión 4.14 en nuestro caso es una multiplicación de todos los valores α_i.

$$P(sob)_i = \left| \prod_{j=1}^{i} \alpha_j \right|^2 . \tag{5.16}$$

Por lo tanto, para calcular la probabilidad de éxito en este caso

$$P(ex)_i = P(mar)_i \cdot P(sob)_i = P(mar)_i \left| \prod_{j=1}^{i} \alpha_j \right|^2 . \tag{5.17}$$

Y si se sustituye la probabilidad de medir el marcado por su valor encuentra el valor de la probabilidad de éxito.

$$P(ex)_i = \frac{S}{N|\prod_{j=1}^{i} \alpha_j|^2} \cdot \left| \prod_{j=1}^{i} \alpha_j \right|^2 = \frac{S}{N} \tag{5.18}$$

Se puede ver que al final la probabilidad de éxito no es superior a hacer la medida directamente en la superposición uniforme $|\phi_0\rangle$.

Esto se puede explicar porque los ángulos que hacen que sea muy probable medir $|\mathbf{0}\rangle$ en el segundo registro hacen también que sea difícil medir el estado marcado como se puede ver en la figura 5.2, de tal forma que se compensa y la probabilidad total no varía.

5.1.1. Simulación

Las gráficas expuestas en la sección 5.1 han sido obtenidas calculando los valores numéricos de las fórmulas (5.11) y (5.15) y representándolos con ayuda de un script de Python. Podemos utilizar esta herramienta informática para hacer una simulación del oráculo y ver qué resultados cabría esperar. En esta simulación se utiliza una librería de Python llama numpy que nos ayuda a hacer cálculos matemáticos. Lo que hace el programa es elegir aleatoriamente S estados del primer registro. Esto se hace eligiendo aleatoriamente S números del 0 al $N-1$. Después iniciamos en un *array* con una longitud N^2 el estado $|\Psi_0\rangle$, (5.3) para un ángulo dado. El siguiente paso es hacer una medida del segundo registro, para ello utilizamos una la función de numpy `np.choice()` que nos permite escoger aleatoriamente un elemento de un *array*, pero teniendo en cuenta las posibilidades de cada elemento. Es decir, a la función le pasamos dos argumentos: el primero son todos los números del 0 al $N^2 - 1$ y el segundo la probabilidad de cada elemento de ser elegido, que viene dada por el módulo al cuadrado de las amplitudes de probabilidad de (5.3).

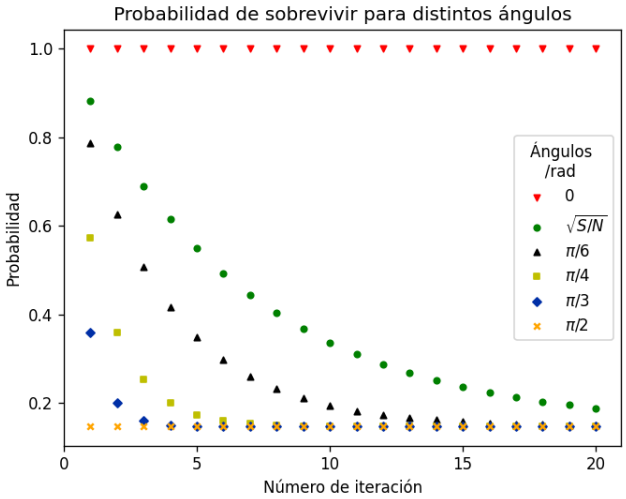


Figura 5.2: Representación de la probabilidad de sobrevivir $|\prod_{j=1}^{i} \alpha_j|^2$ para distintos ángulos.

El siguiente paso es comprobar que hemos "medido". Utilizando el anterior paso se obtiene un número que corresponde a una posición de la representación vectorial del estado $|\Psi_0\rangle$, como de antemano sabemos en que posiciones estarán los estados que correspondan a $|\mathbf{0}\rangle$ del segundo registro, podemos comprobar si el resultado de esta medida es alguno de esas posiciones. En el caso de que hayamos obtenido una medida exitosa el programa directamente aplica el oráculo para obtener el estado $|\Psi_1\rangle$ (5.6) sin pasar por el estado $|\phi_1\rangle$ (5.5).

La aplicación del oráculo consta de distintos pasos, el primero es dotar a todas las posiciones del estado total que no correspondan con un $|\mathbf{0}\rangle$ en el segundo registro de un valor cero. Después calcular el valor de la suma de módulos al cuadrado de las posiciones que quedan (que serán únicamente las correspondientes a los estados $|x\rangle|\mathbf{0}\rangle$). Este valor corresponde al valor $|\alpha_1|^2$ (5.4) por lo que si dividimos el estado total entre la raíz cuadrada de ese valor nos quedará un estado normalizado. En este punto el *array* contiene el estado (5.5) y para conseguir el estado deseado simplemente asignamos a las posiciones correspondientes a $|x\rangle|\mathbf{0}\rangle, x \notin S$ el valor que tenían anteriormente multiplicado por $\cos\theta$ y a las posiciones de $|\mathbf{0}\rangle|x\rangle, x \notin S$ les asignamos el valor correspondiente. De esta forma se pasa del estado $|\phi_1\rangle$ al $|\Psi_1\rangle$. Si queremos realizar otra medida habría que repetir el proceso. Si nos interesa saber cuál es la probabilidad de medir un elemento de los marcados en cierto momento habrá que escoger las posiciones del vector total que corresponden a un elemento marcado y sumar sus módulos al cuadrado.

Gracias a este simulador se pueden comprobar los resultados expuestos en la sección 5.1, calculados analíticamente y las gráficas 5.1. Para ello teniendo en mente el esquema principal que sigue el oráculo y haciendo pequeñas modificaciones al programa se pueden sacar resultados para comprobar.

La primera gráfica que exponemos es 5.3. Está gráfica representa el número de iteraciones necesario para que los estados marcados alcancen una probabilidad de por lo menos el 50 %.

En esta representación entendemos como iteración el número de medidas del segundo registro que serán necesarias para alcanzar la probabilidad dada. Por lo tanto, cuando la barra de la representación 5.3 llegue hasta el número 1 significa que en los estados (5.5) y (5.6) la probabilidad de medir un estado marcado es por lo menos 0.5. Esta gráfica 5.3 se ha realizado obligando a aplicar el oráculo constantemente y comprobando en cada aplicación si la suma de las amplitudes al cuadrado de todos los estados marcados superaba 0.5. Los ángulos elegidos para esta representación son 50 ángulos entre 0 y 2π y equidistantes

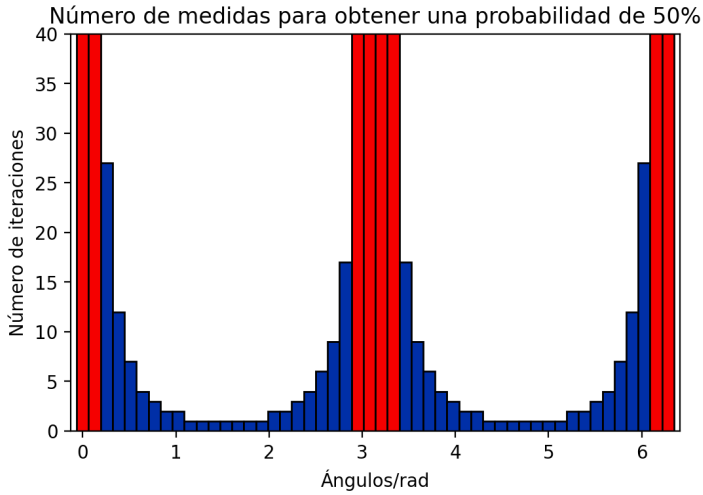


Figura 5.3: Número de iteraciones necesarias para que los estados marcados alcancen una probabilidad de ser medidos del 50 % con $N = 1024$ y $S = 150$.

entre sí.

Se puede ver en la figura 5.3 que para ángulos cercanos a $\pi/2$ se alcanza la probabilidad con solo dos aplicaciones del oráculo. Esto concuerda con lo visto en la gráfica 5.1b para el ángulo $\pi/2$, en la segunda iteración ya tenemos una amplitud muy grande del estado marcado.

Las barras rojas de la figura 5.3 son los ángulos para los cuales no se alcanza una amplitud suficiente después de $\sqrt{N} = 32$ iteraciones. La primera barra corresponde a 0 rad y la última a 2π, para estos ángulos la probabilidad de sobrevivir es 1 ya que la parte del sistema correspondiente a los estados no marcados no tiene ninguna amplitud. Como ya se había comentando, por este mismo motivo, la probabilidad de medir el estado marcado es constante por lo que nunca se va a alcanzar una probabilidad de más del 50 %. Ocurre lo mismo para las dos barras del medio que son cercanas a ángulos de π.

Aprovechando que tenemos el número de iteraciones que hacen falta para que el estado marcado alcance una probabilidad del 50 % podemos simular el oráculo correctamente. Para ello vamos a iniciar el estado (5.3) y hacer una medida, si esta medida resulta en un $|\mathbf{0}\rangle$ del segundo registro tendremos que aplicar otra vez el oráculo. Este paso se repetirá hasta que hayamos sobrevivido al número de medidas dado por la gráfica 5.3. Si en alguna de esas medidas no se obtiene $|\mathbf{0}\rangle$ en el segundo registro el algoritmo habrá fallado por lo que tendremos que volver al paso inicial y empezar de cero, esa iteración no será valida. Suponiendo que hayamos sobrevivido el número de iteraciones dado por la figura 5.3 tendremos que hacer una nueva medida y si esta medida corresponde con un estado marcado el algoritmo habrá resultado exitoso, en caso contrario el algoritmo habrá fallado y se tendrá que iniciar de nuevo. Haciendo 1000 intentos para cada ángulo podremos sacar una buena estadística del porcentaje de veces que el algoritmo resulta exitoso. Si se representan los resultado en un diagrama de barras se obtiene la figura 5.4.

Podemos observar que este porcentaje no depende apenas del ángulo con el que estemos trabajando y las pequeñas variaciones se pueden achacar a variaciones estadísticas. Si en vez de 1000 intentos para cada ángulo lo hubiéramos hecho con 100000 las oscilaciones estadísticas serían menos pronunciadas, pero el tiempo de ejecución del programa crecería enormemente y no sería posible ejecutarlo. Los huecos en la representación 5.4 son los mismos que los huecos correspondientes a las barras rojas de la figura 5.3. Si no se alcanza una probabilidad suficiente en un número de iteraciones razonable, no lo simulamos, de

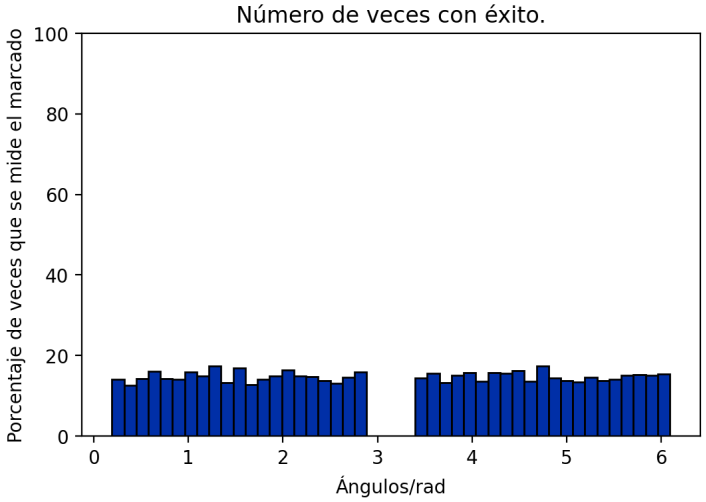


Figura 5.4: Porcentaje de veces que obtenemos un estado marcado con $N = 1024$ y $S = 150$.

esta forma ahorramos tiempo de cálculo.

Con el número de estados que estamos trabajando ($N = 1024$) y el número de elementos marcados ($S = 150$) la probabilidad de medir un elemento marcado directamente en la superposición uniforme (5.2) es $S/N = 0{,}146$, que corresponde dentro de las variaciones estadísticas con la altura que alcanzan las barras en 5.4. Este resultado coincide con el se había pronosticado al final de la sección 5.1 en la expresión (5.18), la probabilidad de éxito no depende del ángulo ni del número de iteraciones, es siempre constante.

Podemos ser un poco más laxos con el número de iteraciones que consideramos válidas. Por ejemplo si permitimos que se aplique el oráculo hasta un máximo de 500 veces la gráfica 5.3 se transforma en la figura 5.5. Aquí se puede comprobar que al igual que antes para el ángulo de 0 y 2π nunca alcanza la probabilidad suficiente y por muchas veces que se aplique el oráculo nunca se va a alcanzar dicha probabilidad. Utilizando está gráfica como referencia se puede extraer la equivalente a 5.4.

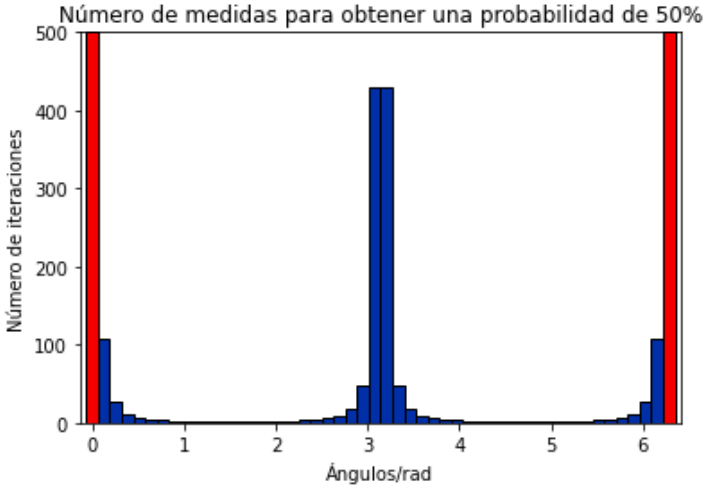


Figura 5.5: Gráfica 5.3 si dejamos evolucionar el sistema más iteraciones.

Como se puede observar no hay ninguna diferencia con la gráfica 5.4 y la probabilidad permanece

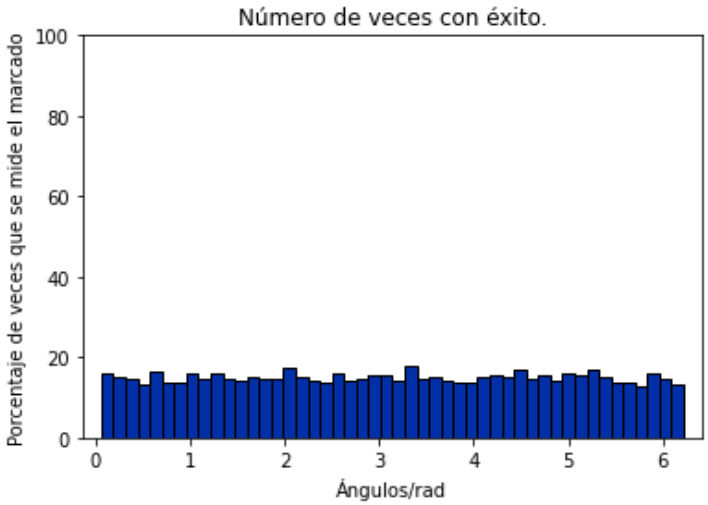


Figura 5.6: Porcentaje de veces que medimos un estado marcado.

constante. Con esto se cumple lo esperado tras la fórmula para calcular la probabilidad de éxito (4.15), la probabilidad es constante. Cabe destacar los distintos tiempos de ejecución que han tenido las dos gráficas. La primera representación 5.4 tardó 1h en ejecutarse mientras que la segunda 5.6 tardó 5h y media.

5.2. Rotación sobre todos los estados

Dado que nuestro primer oráculo no ha aportado se ha hecho un segundo intento. En el primer oráculo no se actuaba sobre los estados marcados y se actuaba en el resto por lo que cabe pensar que para el nuevo oráculo sería necesario actuar sobre ambos. Al igual que siempre se empieza con una superposición uniforme de todos los estados.

$$|\phi_0\rangle = \frac{1}{\sqrt{N}} \sum |x\rangle \,. \tag{5.19}$$

A este estado se le aplica el oráculo.

$$|\Psi_0\rangle = \frac{1}{\sqrt{N}} \sum_{x\in S} \cos\theta\, |\mathbf{0}\rangle\, |x\rangle - \frac{1}{\sqrt{N}} \sum_{x\in S} \sin\theta\, |x\rangle\, |\mathbf{0}\rangle + \frac{1}{\sqrt{N}} \sum_{\substack{x\notin S \\ x\neq 0}} \cos\theta\, |x\rangle\, |\mathbf{0}\rangle + \frac{1}{\sqrt{N}} \sum_{\substack{x\notin S \\ x\neq 0}} \sin\theta\, |\mathbf{0}\rangle\, |x\rangle + \frac{1}{\sqrt{N}} |\mathbf{0}\rangle\, |\mathbf{0}\rangle \,. \tag{5.20}$$

Se puede ver que el efecto de este oráculo en los elementos no marcados es igual al efecto del oráculo de la sección 5.1. Sin embargo también aplica una rotación de los estados marcados. Al igual que antes se realiza una medida del segundo registro y el resultado que interesa es obtener $|\mathbf{0}\rangle$.

$$|\alpha_1|^2 = \frac{S}{N} \sin^2\theta + \frac{NS-1}{N} \cos^2\theta + \frac{1}{N}. \tag{5.21}$$

α_1 es la amplitud de probabilidad de este estado. Esta vez el estado justo después de la medida es:

$$|\phi_1\rangle = \frac{1}{\alpha_1} \left(-\frac{1}{\sqrt{N}} \sum_{x\in S} \sin\theta\, |x\rangle\, |\mathbf{0}\rangle + \frac{1}{\sqrt{N}} \sum_{\substack{x\notin S \\ x\neq 0}} \cos\theta\, |x\rangle\, |\mathbf{0}\rangle + \frac{1}{\sqrt{N}} |\mathbf{0}\rangle\, |\mathbf{0}\rangle \right). \tag{5.22}$$

Y al aplicar otra vez el oráculo se obtiene

$$|\Psi_1\rangle = \frac{1}{\sqrt{N}\alpha_1}\left(-\sum_{x\in S}\cos\theta\sin\theta\,|\mathbf{0}\rangle\,|x\rangle + \sum_{x\in S}\sin^2\theta\,|x\rangle\,|\mathbf{0}\rangle + \sum_{\substack{x\notin S\\ x\neq 0}}\cos^2\theta\,|x\rangle\,|\mathbf{0}\rangle + \sum_{\substack{x\notin S\\ x\neq 0}}\cos\theta\sin\theta\,|\mathbf{0}\rangle\,|x\rangle + |\mathbf{0}\rangle\,|\mathbf{0}\rangle\right). \tag{5.23}$$

El algoritmo tiene los mismos pasos que antes por lo que no hace falta hacer el desarrollo completo. Se puede pasar directamente a comentar cuales serán los estados antes de la $i-$ésima medida.

$$|\phi_{i-1}\rangle = \frac{1}{\prod_{j=1}^{i-1}\alpha_j}\left(\frac{(-1)^{2i-1}}{\sqrt{N}}\sum_{x\in S}\sin^{i-1}\theta\,|x\rangle\,|\mathbf{0}\rangle + \frac{1}{\sqrt{N}}\sum_{\substack{x\notin S\\ x\neq 0}}\cos^{i-1}\theta\,|x\rangle\,|\mathbf{0}\rangle + |\mathbf{0}\rangle\,|\mathbf{0}\rangle\right). \tag{5.24}$$

$$\begin{aligned}|\Psi_{i-1}\rangle &= \frac{1/\sqrt{N}}{\prod_{j=1}^{i-1}\alpha_j}\left((-1)^{2i-1}\sum_{x\in S}\sin^{i-1}\theta\cos\theta\,|\mathbf{0}\rangle\,|x\rangle + (-1)^{2i-2}\sum_{x\in S}\sin^{i}\theta\,|x\rangle\,|\mathbf{0}\rangle\right. \\ &+ \left.\sum_{\substack{x\notin S\\ x\neq 0}}\cos^{i}\theta\,|x\rangle\,|\mathbf{0}\rangle + \sum_{\substack{x\notin S\\ x\neq 0}}\cos^{i-1}\theta\sin\theta\,|\mathbf{0}\rangle\,|x\rangle + |\mathbf{0}\rangle\,|\mathbf{0}\rangle\right).\end{aligned} \tag{5.25}$$

De la expresión (5.25) se puede sacar cuál será la fórmula general para los α_i.

$$|\alpha_i|^2 = \frac{1}{\sqrt{N}|\prod_{j=i}^{i}\alpha_j|^2}\left(\frac{S}{N}\sin^{2i}\theta + \frac{NS-1}{N}\cos^{2i}\theta + 1\right). \tag{5.26}$$

Teniendo esta información ya es posible saber cuál será la probabilidad de medir un estado marcado en el primer registro. Hay que tener en cuenta que al contrario que en la sección 5.1 con este oráculo sí que hay diferencia entre hacer la medida del primer registro en $|\phi_i\rangle$ y en $|\Psi_i\rangle$.

$$|\phi_i\rangle \to P(mar)_i = \frac{S}{N|\prod_{j=i}^{i}\alpha_j|^2}\sin^{2i}\theta. \tag{5.27}$$

$$|\Psi_i\rangle \to P(mar)_i = \frac{S}{N|\prod_{j=i}^{i}\alpha_j|^2}\sin^{2i+2}\theta. \tag{5.28}$$

Se puede observar que medir en el estado $|\Psi_i\rangle$, es decir teniendo la probabilidad (5.28), no es beneficioso. Esto es porque la diferencia entre (5.27) y (5.28) es una multiplicación por $\sin^2\theta$. Sea cual sea el ángulo θ este valor va a ser menor que 1 por lo que al multiplicar el valor obtenido en (5.27) por algo menor que uno se va a obtener algo menor.

Al igual que en la sección anterior la probabilidad de sobrevivir viene dada por la expresión

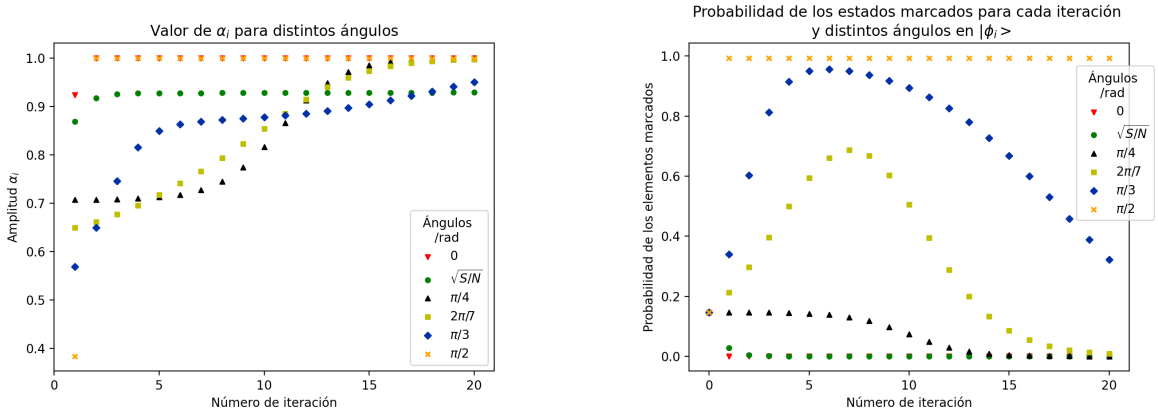
$$P(sob)_i = \left|\prod_{j=1}^{i}\alpha_j\right|^2, \tag{5.29}$$

en este caso cambia el valor de los α_j. Con este nuevo oráculo también se puede definir la probabilidad de éxito como se hizo en la sección 5.1.

$$P(ex)_i = \frac{S}{N|\prod_{j=i}^{i}\alpha_j|^2}\sin^{2i}\theta\left|\prod_{j=1}^{i}\alpha_j\right|^2 = \frac{S}{N}\sin^{2i}\theta. \tag{5.30}$$

Esta probabilidad es claramente peor que la obtenida en (5.18) ya que es el mismo valor multiplicado por algo menor que 1. Pero este oráculo tiene una salvedad, existe la posibilidad de medir un estado marcado al realizar la medida del segundo registro. Por lo tanto hay una diferencia con nuestro anterior algoritmo. En este caso si una de las medidas del segundo registro no da como resultado el estado $|\mathbf{0}\rangle$ habrá que pasar el estado resultante de la medida por el oráculo para identificar si se trata de un estado marcado o no. En la probabilidad de éxito (5.30) habría que tener en cuenta también la posibilidad de haber medido un estado marcado en el caso de que alguna de las medidas del segundo registro no den como resultado el estado $|\mathbf{0}\rangle$.

Con estos resultado se pueden hacer las representaciones gráficas correspondientes a este oráculo.



(a) Valor de los α_i

(b) Representación de $P(mar)_i$ para distintos ángulos

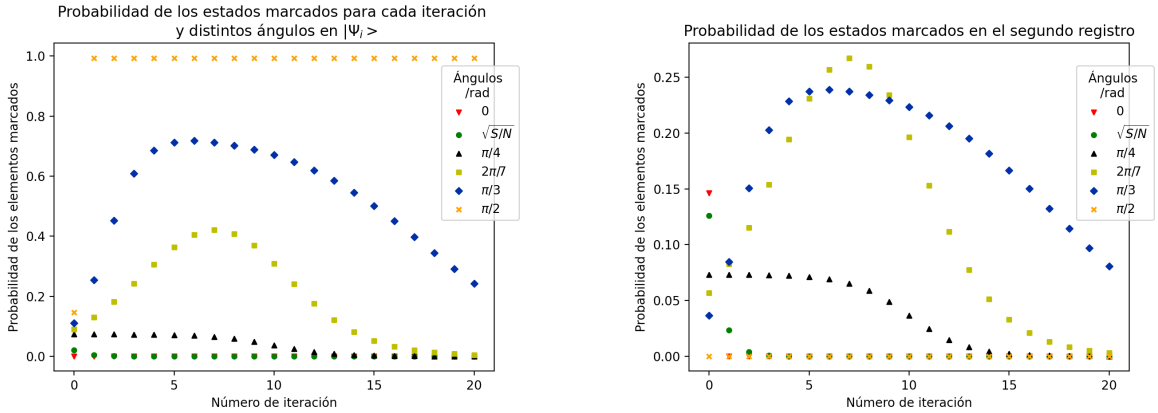
Figura 5.7: Valores de los α_i y la probabilidad (5.27) para distintos ángulos con valores $N = 1024$, $S = 150$.

Se puede ver que las representaciones tienen tendencias distintas a las mostradas en la figura 5.1. En estas figuras se ha cambiado el valor de $\theta = \pi/6$ por $\theta = 2\pi/7$ ya que el primer valor no aportaba información extra y se observaba gran diferencia en la figura 5.7b entre los ángulos $\pi/3$ y $\pi/4$. Al contrario que en la figura 5.1b en 5.7b la probabilidad de medir el estado marcado en el primer registro decrece con las iteraciones. Esto es debido al factor $\sin^{2i}\theta$ ya que como este valor oscila entre 0 y 1 cuantas más veces se multiplique la probabilidad por él más pequeña será. Para los ángulos 0 y $\pi/2$ sigue ocurriendo lo mismo que en 5.1. La representación 5.7b para el resto de ángulos depende mucho del valor de $\sin\theta$, ya que si este valor difiere mucho de 1 la probabilidad de medir un estado marcado cae rápidamente.

Puede ser interesante mostrar dos representaciones más que no eran posibles con el primer oráculo.

La primera de las gráficas es la probabilidad de medir un estado marcado en el estado $|\Psi_i\rangle$ que se comentó que debía ser igual o peor para cualquier ángulo que la probabilidad de medir en $|\phi_i\rangle$. Con la representación 5.8a se comprueba. En la segunda representación se comprueba la probabilidad de medir un estado marcado en el segundo registro. Esta probabilidad viene dada por la expresión

$$P(mar_2)_i = \frac{S\sin^{2i}\theta\cos^2\theta}{N|\prod_{j=i}^{i}\alpha_j|^2}. \tag{5.31}$$



(a) Probabilidad (5.28)

(b) Marcados en el segundo registro

Figura 5.8: Probabilidad de medir el estado marcado en el estado $|\Psi_i\rangle$ y en el segundo registro con valores $N = 1024$ y $S = 150$.

Se observa una tendencia similar a la vista en 5.7b dado que la dependencia con el ángulo θ es parecida. Aunque cabe destacar que la máxima probabilidad alcanzada es poco mayor de 0.25. Por último, mostrar la gráfica de la probabilidad de sobrevivir en la figura 5.9. Esta representación sigue una tendencia igual a la figura 5.2, pero la probabilidad de sobrevivir con este oráculo es menor que con el anterior.

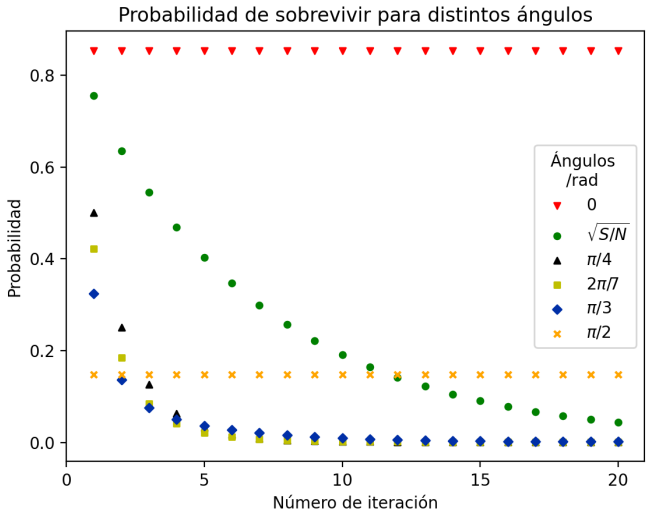


Figura 5.9: Representación de la probabilidad de sobrevivir (5.29) para distintos ángulos.

5.2.1. Simulación

La simulación para este oráculo es muy similar a la que ese utilizó para el anterior oráculo. Por lo tanto no se va a volver a explicar y se van a comentar las diferencias.

En la primera simulación si al hacer una medida del segundo registro no se obtenía $|\mathbf{0}\rangle$ el algoritmo se detenía; en esta simulación si ocurre esto se verifica si el resultado de la medida es un estado marcado

antes de detener la simulación. Como con este oráculo sí que hay diferencia entre medir el primer registro en $|\phi_i\rangle$ y $|\Psi_i\rangle$ hay que crear una función que dé como resultado el estado $|\phi_i\rangle$. Para ello la función toma como argumentos el *array* que contiene el estado total y las posiciones de ese *array* que no equivalen a un estado cuyo segundo registro sea $|\mathbf{0}\rangle$. Asigna a estas posiciones el valor 0 y después suma los módulos al cuadrado del resto de posiciones. Por último, se normaliza el *array* dividiéndolo entre el valor obtenido antes. La función que nos da los estados $|\Psi_i\rangle$ funciona de la misma forma que la utilizada en la sección 5.1.1, por lo que no se va a volver a explicar.

Este oráculo tiene una dependencia con los ángulos más brusca como se puede ver en la figura 5.7b en consecuencia las gráficas se han acortado. En vez de representar 50 ángulos entre 0 y 2π se representan los resultados para 50 ángulos entre 0 y π ya que se puede observar que las gráficas son simétricas para estos ángulos.

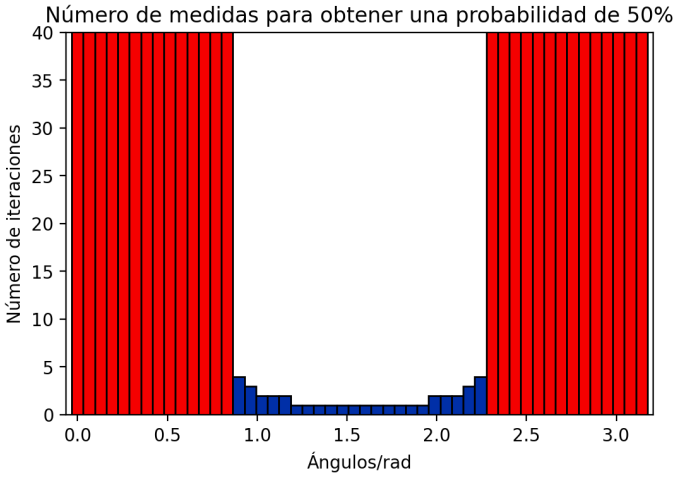


Figura 5.10: Número de iteraciones hasta conseguir que los estados marcados tengan una probabilidad mayor del 50 % para el segundo oráculo.

Esta sería la equivalente a la figura 5.3 para este oráculo. Se puede observar que hay mucha más presencia de barras rojas que identificaban los ángulos para los que los estados marcados no alcanzan una probabilidad mayor del 50 % antes de $\sqrt{N} = 32$ iteraciones. La presencia de tantas lineas rojas se explica por la expresión (5.27). Cuanto más se aleje el valor de $\sin\theta$ de 1 más pequeña será la probabilidad ya que habrá que multiplicar numerosas veces por algo menor que 1. Por ejemplo la última barra roja del bloque de la izquierda corresponde con un ángulo de $\theta = 0{,}833$ rad y $\sin\theta = 0{,}74$. Después de 6 iteraciones el valor que adquiere el término seno de (5.27) vale $\sin^{12}\theta = 0{,}026$ por lo que es muy difícil que se adquiera una probabilidad mayor del 50 %. Esta representación se ha obtenido aplicando el oráculo repetidamente y comprobando después de cada medida si la amplitud de probabilidad de los elementos marcados en el estado $|\Psi_i\rangle$ al cuadrado era mayor de 0.5. En caso de que este valor no se obtuviera para 32 iteraciones se detenía la simulación.

Utilizando los resultados de la gráfica 5.10 se puede realizar una simulación del algoritmo real. Al contrario que en la sección 5.1.1, en esta ocasión se tendrán en cuenta todos los ángulos, independientemente de si se alcanza una probabilidad de 0.5 o no, ya que existe la posibilidad de medir un elemento marcado en el segundo registro como se mostró en la figura 5.8b.

Se observa que el porcentaje de veces que se obtiene un estado marcado no depende del ángulo como sucedía con el primer oráculo. Además, vuelve a ocurrir que no se gana ninguna probabilidad con respecto al estado inicial de superposición uniforme $|\phi_0\rangle$. Si se define la probabilidad de éxito como se hizo en el

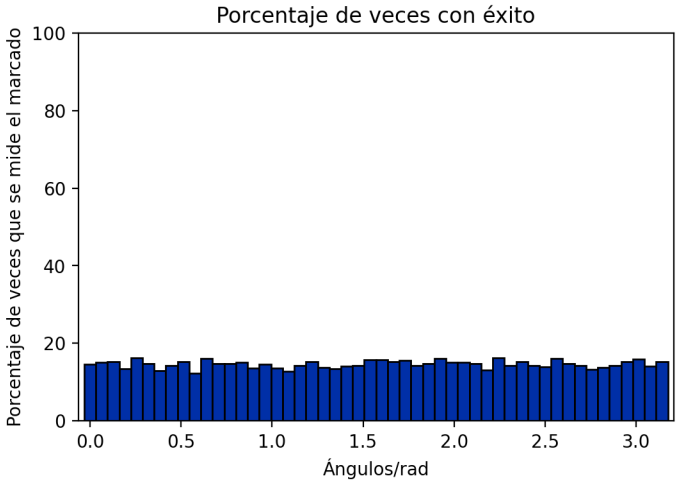
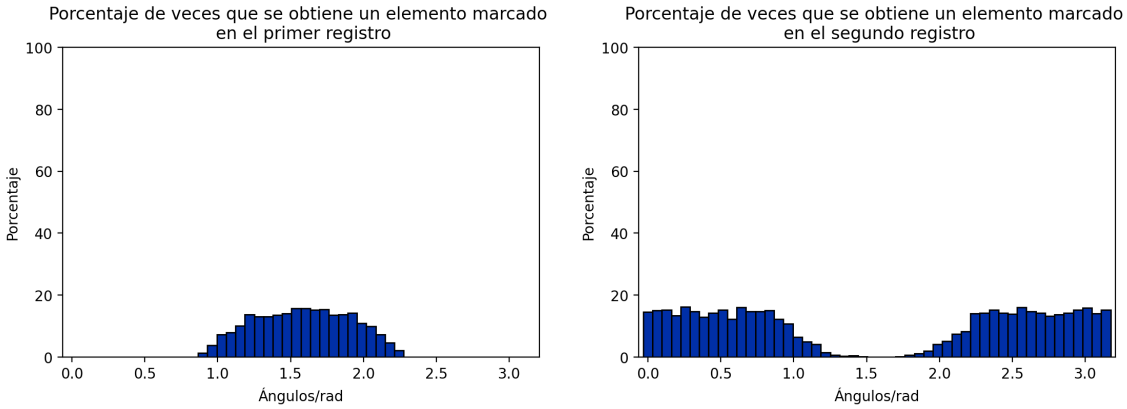


Figura 5.11: Número de veces que se mide el estado marcado por cada 100 iteraciones.

anterior oráculo cabría esperar ver cierta diferencia para los ángulos.

$$P(ex)_i = \frac{S \sin^{2i} \theta}{N}. \tag{5.32}$$

Pero se ha de tener en cuenta la probabilidad de medir el estado marcado en el segundo registro. Como se puede ver en la representación 5.12 estas dos probabilidades se complementan haciendo que cuando es poco probable medir el estado marcado en el primer registro la amplitud de los estados $|\mathbf{0}\rangle |x\rangle |x \in S$ crece.



(a) Medida de marcado en el primer registro. (b) Medida de marcado en el segundo registro.

Figura 5.12: Número de veces en las que se mide un estado marcado en el primer registro y en el segundo por cada 100 medidas.

Capítulo 6

Conclusiones

Por finalizar este trabajo se va a hacer un breve repaso de lo expuesto en él y se comentarán las conclusiones que se pueden extraer.

1. En este trabajo se ha estudiado el algoritmo de Grover desde varios puntos de vista. Estos puntos de vista difieren entre sí en la manera en la que se consigue hacer una transferencia de amplitud desde unos estados hasta otros, pero los resultados que obtienen son los mismos. Los puntos de vista analizados son:

   a) Un símil con un sistema físico como es un pozo de potencial. En la sección 3.1 se dedujo el algoritmo de Grover a partir de la ecuación de Schrödinger y se mostró el primer acercamiento a este algoritmo.

   b) Una versión más formal en la que se visualiza de manera clara el funcionamiento de este algoritmo, presentada en la sección 3.2.

   c) Una visión geométrica del algoritmo que muestra las evoluciones elementales como rotaciones en un plano bidimensional, mostrada en la sección 4.1.

2. Partiendo de este último punto de vista se aporta una descripción teórica de la realización del algoritmo en términos de una secuencia de medidas. Para dar este punto de vista nos apoyamos en el efecto Zenón cuántico explicado en la sección 4.2. Pero en vez de utilizarlo para "congelar"la evolución de un sistema, se utiliza para causarla.

3. En el capítulo 5 se exploran dos intentos de definir una medida que pueda darnos la evolución del algoritmo de Grover. Estos dos intentos tienen limitaciones que vienen dadas por varios hechos:

   a) No se puede actuar únicamente en la parte del estado que no nos interesa, ya que lo que se consigue así es variar la amplitud de probabilidad de los elementos no marcados y dejar igual la de los elementos marcados, que únicamente varía debido a las normalizaciones que hay que hacer después de cada medida. De este modo cuando se tiene en cuenta la probabilidad de sobrevivir los dos efectos se contrarrestan y no se obtiene mejora alguna con respecto a la medida directa en el estado inicial $|\phi_0\rangle$.

   b) Si se actúa por igual sobre todos los elementos lo que se logra es dividir la probabilidad de medir un estado marcado entre los dos registros que se tienen lo que causa que tampoco se obtenga mejora con respecto del primer intento.

La principal limitación que tienen los algoritmos cuánticos cuya evolución viene dada por medidas sucesivas es que no se puede controlar con total exactitud el resultado que se va a obtener al hacer dicha medida. Por lo tanto al calcular la probabilidad de éxito final hay que tener en cuenta la probabilidad de que todas las medidas realizadas para la evolución del sistema hayan resultado correctas, como acabamos de explicar. Si se realiza la transferencia de probabilidades únicamente con medidas lo más seguro es que el efecto se acabará contrarrestando.

Aunque tal y como está expuesto en la sección 4.3 es teóricamente posible, la implementación practica presenta más dificultades. Aunque se encuentre una implementación correcta tal y como se muestra en las gráficas 4.4 el algoritmo controlado por medidas es menos eficiente que el usual.

Si se quiere seguir el mismo esquema seguido en este trabajo se podría crear un algoritmo híbrido que combine las medidas sucesivas con la utilización de un operador de difusión que actúe sobre el estado después de cada medida. Con esto se conseguiría que la difusión de probabilidades no dependa exclusivamente de medidas sucesivas.

Bibliografía

[1] Michael A. Nielsen e Isaac L. Chuang. *Quantum Computation and Quantum Information*. 10th anniversary ed. Cambridge ; New York: Cambridge University Press, 2010. 676 págs. ISBN: 978-1-107-00217-3.

[2] Scott Aaronson. «The Limits Of Quantum». En: *SCIENTIFIC AMERICAN* 298.3 (2008), págs. 62-69. ISSN: 00368733, 19467087. JSTOR: 10.2307/26000518.

[3] Claude Cohen-Tannoudji, Bernard Diu y Franck Laloë. *Quantum Mechanics. Volume 1: Basic Concepts, Tools, and Applications*. Trad. por Susan Reid Hemley, Nicole Ostrowsky y Dan Ostrowsky. Second edition. Weinheim: Wiley-VCH Verlag GmbH & Co. KGaA, 2020. 921 págs. ISBN: 978-3-527-34553-3.

[4] Lov K. Grover. «Quantum Mechanics Helps in Searching for a Needle in a Haystack». En: *Physical Review Letters* 79.2 (14 de jul. de 1997), págs. 325-328. ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.79.325. URL: https://link.aps.org/doi/10.1103/PhysRevLett.79.325.

[5] Peter W. Shor. «Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer». En: *SIAM review* 41.2 (1999), págs. 303-332.

[6] Lov K. Grover. «From Schrödinger's Equation to the Quantum Search Algorithm». En: *Pramana* 56.2-3 (feb. de 2001), págs. 333-348. ISSN: 0304-4289, 0973-7111. DOI: 10.1007/s12043-001-0128-3. arXiv: quant-ph/0109116. URL: http://arxiv.org/abs/quant-ph/0109116.

[7] Michel Boyer y col. «Tight Bounds on Quantum Searching». En: *Fortschritte der Physik* 46.4-5 (jun. de 1998), págs. 493-505. ISSN: 0015-8208, 1521-3978. DOI: 10.1002/(SICI)1521-3978(199806)46:4/5<493::AID-PROP493>3.0.CO;2-P. arXiv: quant-ph/9605034. URL: http://arxiv.org/abs/quant-ph/9605034.

[8] David Deutsch y Richard Jozsa. «Rapid Solution of Problems by Quantum Computation». En: *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences* 439.1907 (8 de dic. de 1992), págs. 553-558. DOI: 10.1098/rspa.1992.0167. URL: https://royalsocietypublishing.org/doi/10.1098/rspa.1992.0167.

[9] N. David Mermin. *Quantum Computer Science: An Introduction*. Cambridge: Cambridge University Press, 2007. ISBN: 978-0-511-81387-0. DOI: 10.1017/CBO9780511813870. URL: http://ebooks.cambridge.org/ref/id/CBO9780511813870.

[10] Paul Kwiat, Harald Weinfurter y Anton Zeilinger. «Quantum Seeing in the Dark». En: *Scientific American* 275.5 (nov. de 1996), págs. 72-78. ISSN: 0036-8733. DOI: 10.1038/scientificamerican1196-72. URL: https://www.scientificamerican.com/article/quantum-seeing-in-the-dark.

[11] Wayne M. Itano y col. «Quantum Zeno Effect». En: *Physical Review A* 41.5 (1 de mar. de 1990), págs. 2295-2300. DOI: 10.1103/PhysRevA.41.2295. URL: https://link.aps.org/doi/10.1103/PhysRevA.41.2295.