



Grado en Física  
Trabajo de fin de grado

## Algoritmos Cuánticos

Dorye L. Esteras Córdoba

---

Director:  
David Zueco Láinez

---

Curso 2017-2018



# Índice general

0.1. Introducción . . . . .	1
0.2. Objetivos . . . . .	1
0.3. Desarrollo del trabajo . . . . .	2
<b>1. Conceptos de mecánica cuántica</b>	<b>3</b>
1.1. Formalismo de la Matriz Densidad . . . . .	3
1.1.1. Introducción del operador densidad . . . . .	3
1.1.2. Traza parcial: Descripción separada de una parte de un sistema físico . . . . .	5
1.2. Entrelazamiento y mezcla. . . . .	6
1.3. Decoherencia . . . . .	7
1.3.1. Marco del problema . . . . .	7
1.3.2. Modelo de Phase Damping Channel . . . . .	7
<b>2. Computación cuántica</b>	<b>9</b>
2.1. Introducción a la computación cuántica . . . . .	9
2.1.1. ¿Por qué un ordenador cuántico? . . . . .	9
2.1.2. Concepto de qubit . . . . .	10
2.1.3. Puertas Cuánticas . . . . .	11
2.2. Implementación física de puertas: Oscilaciones de Rabi . . . . .	13
2.2.1. Puertas de 1 qubit . . . . .	13
2.2.2. Puertas de 2 qubits . . . . .	15
<b>3. IBM Quantum experience</b>	<b>17</b>
3.1. ¿Qué es IBM Quantum Experience? . . . . .	17
3.2. Dentro de la Quantum Experience . . . . .	17
3.3. Un sistema para usar mecánica cuántica en un ordenador . . . . .	19
<b>4. Resultados obtenidos de un ordenador cuántico</b>	<b>21</b>
4.1. Estudio de la decoherencia . . . . .	21
4.2. Implementación y simulación de un puerta Toffoli . . . . .	23
4.3. Conclusiones . . . . .	25



## **Agradecimientos**

Quiero agradecer a David Zueco, el haber diseñado este tfg, dándome la oportunidad de acceder al uso de un ordenador cuántico real y de realizar un trabajo de fin de grado que, alejado de ser un trabajo bibliográfico, me ha hecho sentir como un pequeño investigador en todo momento. Gracias por tu ayuda y paciencia.

También me gustaría agradecer a Jorge Calvo su interés y su ayuda siempre que la he necesitado, así como la participación de Fernando Luis Vitalla.



## 0.1. Introducción

Este trabajo está realizado dentro del marco de la mecánica cuántica, y supone una base para introducirse en el mundo de la computación. Un ordenador cuántico presenta una serie de propiedades como el paralelismo, que pueden ser cruciales para abordar desde una nueva perspectiva problemas que resultan irresolubles, dejando a un lado las limitaciones de la tecnología clásica. La criptografía cuántica es una de las aplicaciones de mayor interés, que daría lugar a comunicaciones completamente seguras donde cualquier espía sería detectado como consecuencia directa del teorema de no clonado. La construcción de un ordenador cuántico es el principal reto dentro de la computación cuántica, importantes empresas como IBM y Google han entrado a participar en el desarrollo de ordenadores que sigan las leyes de la mecánica cuántica, y que sean capaces de utilizar fenómenos como la superposición y el entrelazamiento. Todo esto ha concluido en la creación de IBM Quantum Experience, un proyecto realizado por la empresa IBM, donde prototipos de ordenadores cuánticos han sido abiertos al público, hecho que es en sí mismo, el eje central de este trabajo. Una de las ideas que busca transmitir IBM es la necesidad de crear un compromiso entre la mejora de la fidelidad y del número de qubits. Existen diversos factores que mejorar en el desarrollo de un ordenador cuántico, pero sin duda el mayor reto es mantener la calidad en sistemas de orden mayor y por lo tanto, construir ordenadores con una mayor cantidad de qubits, ya que al fin y al cabo, está es una condición necesaria para poder implementar cualquier algoritmo útil. No obstante la mejora de la fidelidad es fundamental para obtener unos resultados correctos de estos algoritmos, existen diversos tipos de error, principalmente tres: el error correspondiente a los estados, el error debido a las puertas y el error en la medida. El intento de mejorar todas estas fuentes de error esta siendo llevado a cabo desde muy distintos ámbitos de la física, tanto teórica como experimental. Por este mismo motivo, una parte importante de este trabajo consiste en entender los procesos que hay detrás de la aparición del error y obtener resultados de ordenadores cuánticos reales que los ilustren.

## 0.2. Objetivos

- Comprender los principios básicos de la computación cuántica
- Aprender qué es la computación cuántica, sus objetivos y sus limitaciones.
- Entender qué son puertas lógicas cuánticas y su implementación.
- Aprender a realizar simulaciones en un ordenador cuántico real.
- Uso de algoritmos cuánticos para obtener conclusiones sobre las capacidades de un ordenador cuántico real.

### 0.3. Desarrollo del trabajo

La memoria esta estructurada en cuatro capítulos. En el primero hago una explicación de conceptos de mecánica cuántica básicos, en concreto, introduzco el operador densidad, y los fenómenos de superposición y entrelazamiento. Para concluir explico el proceso de decoherencia y presento un modelo para ilustrarlo. En el segundo capítulo, comienzo con una introducción a la computación cuántica, presento conceptos básicos como el qubit y las principales puertas lógicas y finalmente hago una explicación de las oscilaciones de Rabi y de como tiene lugar la implementación de las puertas lógicas cuánticas. En el tercer capítulo presento IBM Quantum experience y los ordenadores con los que he trabajado. Por último en el cuarto capítulo, acabo con los objetivos del trabajo haciendo una presentación de los resultados obtenidos en los ordenadores cuánticos reales, en concreto, el estudio de la decoherencia mediante el desarrollo de un algoritmo sencillo y el análisis de los resultados obtenidos en la implementación de un algoritmo de implementación de la puerta Toffoli.

# Capítulo 1

## Conceptos de mecánica cuántica

### 1.1. Formalismo de la Matriz Densidad

#### 1.1.1. Introducción del operador densidad

Comienzo este trabajo presentando una herramienta que me ha resultado esencial a la hora de trabajar con estados cuánticos, el operador densidad. Hasta este momento, siempre había trabajado conociendo por completo el estado de las partículas, utilizando funciones de onda perfectamente definidas. En este marco, contaba únicamente con la incertidumbre de la medida, propia de la mecánica cuántica, donde el valor obtenido en la medición de un observable es probabilístico. Sin embargo, durante este trabajo me ha resultado necesario aprender a incorporar, la incertidumbre debida a la información incompleta sobre la función de onda que describe el estado de las partículas.

De esta manera, una partícula puede encontrarse en una mezcla estadística de estados  $\Psi_k$ , cada uno con una determinada probabilidad  $p_k$ , y en cada estado, la medida de un observable nos puede dar un valor distinto con una determinada probabilidad. Es así como a la incertidumbre de la medida se le añade la incertidumbre de no tener un estado perfectamente definido. A partir de ahora, haré una distinción entre los estados perfectamente definidos, llamados puros, y los no puros, también llamados estados mixtos o estados mezcla.

Para lidiar con este nuevo problema, se introduce el operador densidad  $\rho$ , el cual se puede definir fácilmente para estados puros a partir de la definición de valor medio. Haciendo uso del álgebra básica de la mecánica cuántica el valor medio de un operador se puede expresar en una base a través de las coordenadas:

$$|\Psi(t)\rangle = \sum_n c_n(t)|u_n\rangle \quad (1.1)$$

$$\langle A\rangle(t) = \langle\Psi(t)|A|\Psi(t)\rangle = \sum_{n,p} c_n^*(t)c_p(t)\langle u_n|A|u_p\rangle = \sum_{n,p} c_n^*(t)c_p(t)A_{np} \quad (1.2)$$

Podemos expresar estas coordenadas teniendo en cuenta que los coeficientes no son más que la proyección de los estados sobre la base:

$$c_n^*(t)c_p(t) = \langle\Psi(t)|u_n\rangle\langle u_p|\Psi(t)\rangle = \langle u_p|\Psi(t)\rangle\langle\Psi(t)|u_n\rangle \quad (1.3)$$

De donde podemos extraer la definición del operador densidad:

$$\rho(t) = |\Psi(t)\rangle\langle\Psi(t)| \quad (1.4)$$

Y su expresión matricial:

$$\rho_{pn} = \langle u_p | \rho(t) | u_n \rangle = c_n^*(t) c_p(t) \quad (1.5)$$

En el caso mixto, la expresión del operador debe incluir las distintas funciones de onda que describen el estado y sus respectivas probabilidades, por lo que su expresión es:

$$\rho(t) = \sum_k p_k \rho_k \quad (1.6)$$

Donde  $\rho_k$  es  $|\Psi_k(t)\rangle\langle\Psi_k(t)|$ , como se mostraba en la ecuación (1.4)

### Propiedades generales del operador densidad

Enuncio aquí las propiedades principales del operador densidad para el caso de estados puros, que como mostraré después, son análogas a las del caso mixto exceptuando que en este último, el operador no será un proyector.

En primer lugar, el operador densidad es un operador hermítico, cumpliéndose:

$$\rho(t) = \rho^\dagger(t) \quad (1.7)$$

Cuya matriz es definida positiva:

$$\rho(t) > 0 \quad (1.8)$$

Además, es el mismo para estados que difieren únicamente en un factor de fase constante:

$$|\Psi(t)\rangle, e^{i\theta}|\Psi(t)\rangle \longrightarrow \rho(t) \quad (1.9)$$

Nos permite obtener el valor medio de un operador, desarrollando la ecuación (1.2):

$$\langle A \rangle(t) = \sum_{n,p} \langle u_p | \rho(t) | u_n \rangle \langle u_n | A | u_p \rangle = \sum_p \langle u_p | \rho(t) A | u_p \rangle \quad (1.10)$$

De donde podemos identificar la propiedad siguiente:

$$\langle A \rangle(t) = \text{Tr}[\rho(t)A] \quad (1.11)$$

Por último existe una propiedad importante relacionada con la traza de este operador:

$$\text{Tr}[\rho(t)] = 1 \quad (1.12)$$

Esta se puede deducir fácilmente para el caso puro analizando la condición de conservación de la probabilidad:

$$\sum_n |c_n(t)|^2 = 1 = \sum_n \rho_{nn}(t) = \text{Tr}[\rho(t)] \quad (1.13)$$

Para el caso mixto, no hay más que observar que:

$$\text{Tr}[\rho(t)] = \sum_k p_k \text{Tr}[\rho_k(t)] = 1 \quad (1.14)$$

Donde se ha tenido en cuenta que la traza de  $\rho_k$  es 1 y que la suma de las probabilidades  $p_k$  es 1.

### Propiedades no generales

Todas las propiedades antes mostradas son comunes para el caso puro y mixto, no obstante, el caso puro es un caso particular en el que el operador densidad es un proyector, como se puede observar en la expresión 1.4. Esto significa que un estado puro cumple la propiedad:

$$\rho^2 = \rho \quad (1.15)$$

Y por tanto que

$$\text{Tr}[\rho^2] = \text{Tr}[\rho] = 1 \quad (1.16)$$

Además de que su espectro de autovalores estará formado por  $\lambda = 1$  y  $\lambda = 0$ .

Todo esto no es válido en estados que no son puros, en los cuales  $\rho^2 \neq \rho$ . De hecho, para estos podemos deducir otra propiedad. Como la traza de  $\rho$  es 1, sus elementos de la diagonal serán menores que uno, excepto en el caso puro que tendrá ceros y un 1 (proyector). Los valores de la diagonal de  $\rho^2$  serán entonces números menores que 1 elevados al cuadrado, excepto en el especial caso del estado puro, que serán de nuevo ceros y un 1. Este razonamiento concluye entonces, en que la traza será siempre menor o igual que 1, donde este último caso, corresponde al de un estado puro (caso particular de los estados mezcla), es por esto que a la traza de  $\rho^2$  se le llama pureza.

$$\text{Tr}[\rho^2] \leq 1 \quad (1.17)$$

#### 1.1.2. Traza parcial: Descripción separada de una parte de un sistema físico

Suponemos dos sistemas,  $\epsilon(1)$  y  $\epsilon(2)$ , cuyas bases son  $|u_n(1)\rangle$  y  $|v_p(2)\rangle$  respectivamente. Ambos sistemas, son subsistemas del espacio total  $\epsilon = \epsilon(1) \otimes \epsilon(2)$ . Dentro de estas consideraciones, habrá una matriz densidad para el espacio  $\epsilon$ , que contenga la información de ambos subsistemas. La traza parcial, permite obtener la matriz densidad de estos subsistemas y trabajar con ellos aisladamente, sin tener en cuenta toda la información contenida en el sistema total. De esta forma se pueden obtener los elementos de matriz de  $\rho(1)$ , que es la matriz densidad correspondiente al subsistema 1, simplemente contrayendo los indices del sistema 2, en la expresión del operador densidad del sistema total:

$$\begin{aligned} \langle u_n(1) | \rho(1) | u_{n'}(1) \rangle &= \sum_p \langle u_n(1) | \langle v_p(2) | \rho | u_{n'}(1) \rangle | v_p(2) \rangle = \\ &= \langle u_n(1) | \rho(1) | u_{n'}(1) \rangle \sum_p \langle u_p(2) | \rho(2) | u_p(2) \rangle \end{aligned} \quad (1.18)$$

Donde  $\rho$  es el operador densidad del sistema total y donde se reconoce la suma como la traza sobre el sistema 2:

$$\rho(1) = \text{Tr}_2[\rho] \quad (1.19)$$

análogamente podemos obtener:

$$\rho(2) = \text{Tr}_1[\rho] \quad (1.20)$$

Aquí queda manifiesta la propiedad antes mencionada de obtener la información de un subsistema a partir del sistema total. Podemos trasladar esta propiedad al cálculo de valores medios de observables:

Un operador  $A(1)$  que actúa en el espacio  $\epsilon(1)$  puede extenderse a un operador  $\tilde{A}(1) = A(1) \otimes I(2)$  que actúa en  $\epsilon$ . Si tenemos entonces un operador del espacio total como  $\tilde{A}(1)$ , que solo actúa en un subespacio, podemos obtener su valor medio a través de la traza parcial:

$$\langle \tilde{A}(1) \rangle = \text{Tr}(\rho(1)A(1)) \quad (1.21)$$

## 1.2. Entrelazamiento y mezcla.

En este apartado introduzco los conceptos de superposición y entrelazamiento, que son las principales propiedades en las que se basa la computación cuántica. Por otro lado, pretendo introducir la relación entre entrelazamiento y pérdida de pureza; Estudiaré más adelante este punto con un modelo sencillo, para ilustrar como este proceso de perdida de pureza es el principal problema a la hora de desarrollar un ordenador cuántico.

Generar superposición consiste en crear un estado que es combinación lineal de otros. Por ejemplo, en un sistema de dos niveles podría obtenerse la siguiente superposición:

$$|\Psi\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle \quad (1.22)$$

Como explicaré más adelante, la superposición permite generalizar la unidad básica de un ordenador, el bit, que puede tomar valores 0 y 1, al qubit, que contiene a todos los estados que sean combinación lineal de estos dos.

El entrelazamiento es un fenómeno que involucra a varios qubits. Estos pueden entrelazarse dando lugar a un nuevo estado, que se caracteriza por no poder escribirse como producto tensorial de los estados anteriores. Por ejemplo, en un sistema de dos qubits se podría obtener un estado con la siguiente forma:

$$|\Psi\rangle = \frac{1}{\sqrt{2}}|0_A\rangle|0_B\rangle + \frac{1}{\sqrt{2}}|1_A\rangle|1_B\rangle \quad (1.23)$$

El entrelazamiento tiene algunas consecuencias muy interesantes, en primer lugar un estado entrelazado tiene la propiedad de que si realizamos una medida sobre un qubit, la medida sobre el otro qubit queda completamente determinada. Por ejemplo, en el caso descrito en (1.23), al medir A, el valor de B queda completamente determinado, de forma que si se obtiene  $|0\rangle$ , el valor de B es  $|0\rangle$  y si se obtiene  $|1\rangle$ , el valor de B es  $|1\rangle$ .

Otra característica interesante es que una matriz densidad de dos estados que están entrelazados, nos dará tras realizar una traza parcial, estados mixtos, independientemente de que fueran

puros antes de entrelazarse. En el ejemplo (1.23), dos estados puros ( $|0\rangle, |1\rangle$ ) se entrelazaban dando lugar a un estado final, la traza parcial de este estado nos devolverá una matriz de un estado mixto. En la siguiente tabla hago un resumen de los distintos comportamientos de la pureza:

Sistema total	subsistema (traza parcial)
Sistema de estados producto	Estados puros
Sistema de dos estados mixtos	Estados mixtos
Sistema de dos estados puros que se entrelazan	Estados se vuelven mixtos

Además, el estado del ejemplo (1.23), es un estado maximamente entrelazado y por lo tanto, al realizar la traza parcial se obtendrá una representación proporcional a la matriz identidad.

### 1.3. Decoherencia

#### 1.3.1. Marco del problema

La interacción con el exterior, da lugar a procesos de decoherencia mediante los cuales, los estados cuánticos se transforman en estados mezcla, transformando al sistema en un sistema clásico. Este es el gran problema que tienen los ordenadores cuánticos en la actualidad, deben mantenerse en un medio que los vuelve cada vez más clásicos. Trataré de ilustrar este proceso de decoherencia usando un modelo sencillo, que nos permitirá además, comprender porqué no observamos con frecuencia fenómenos cuánticos en la naturaleza macroscopica.

#### 1.3.2. Modelo de Phase Damping Channel

Supongamos una partícula que puede encontrarse inicialmente en dos estados,  $x_0$  y  $-x_0$ . La partícula no está completamente aislada, ya que el aislamiento nunca es perfecto, e interactúa con los fotones que le llegan. Por sencillez del modelo, suponemos también que la interacción tiene lugar únicamente con un fotón y que la masa de la partícula es lo suficientemente grande como para no variar su posición al interactuar con él.

De esta forma, tenemos un sistema donde un fotón puede interactuar con la partícula con una probabilidad  $P$ , acabando en un estado que llamaré  $|1\rangle$  o no interactuar acabando en un estado  $|0\rangle$ . La función de onda que describe el proceso será la siguiente:

$$|\Psi_{x_0}\rangle = \sqrt{P}|x_0\rangle|1\rangle + \sqrt{1-P}|x_0\rangle|0\rangle \quad (1.24)$$

Análogamente si la partícula comienza en la posición  $-x_0$ , el fotón llegará a un estado  $|0\rangle$  en caso de no interacción. No obstante, si se produce interacción, el estado del fotón después de la interacción no será  $|1\rangle$ , será un estado distinto debido a las distintas condiciones iniciales, llamaré a este estado final  $|2\rangle$ .

Teniendo en cuenta esto, la función de onda representativa del proceso será:

$$|\Psi_{-x_0}\rangle = \sqrt{P}|-x_0\rangle|2\rangle + \sqrt{1-P}|-x_0\rangle|0\rangle \quad (1.25)$$

Si el experimento comienza con la partícula en  $x_0$ , se tendrá la primera función de ondas, si inicialmente la partícula está en  $-x_0$ , se tendrá la segunda, de igual forma, si el sistema comienza en una superposición de ambos estados, la función de ondas que describirá el proceso será una superposición de las funciones de ondas anteriores:

$$|\Psi_{inicial}\rangle = \frac{1}{\sqrt{2}}(|x_0\rangle + |-x_0\rangle)|0\rangle \quad (1.26)$$

Cuya evolución temporal dará lugar a la siguiente función de onda:

$$|\Psi_{final}\rangle = \frac{1}{\sqrt{2}}(\sqrt{P}(|x_0\rangle|1\rangle + |-x_0\rangle|2\rangle) + \sqrt{1-P}(|-x_0\rangle + |x_0\rangle)|0\rangle) \quad (1.27)$$

Este estado no puede escribirse como producto tensorial de sus estados iniciales, es un estado entrelazado, como se explicó en la sección 1.2. Ahora bien, ya que solo estamos interesados en la partícula, la medida de interés, es la realizada sobre el espacio de la partícula, no sobre el espacio total. La traza parcial nos permite obtener la información que buscamos, sin embargo, como he comentado en una parte previa de esta memoria, la traza parcial de la matriz densidad de un estado entrelazado, tiene como resultado una matriz densidad de un estado mixto. Analizo con más detalle este punto, nuestro estado inicialmente era un estado puro, pero al interactuar con el fotón se ha hecho mixto. El estado inicial de la partícula era una superposición:

$$|\Psi_{partcula}\rangle = \alpha|x_0\rangle + \beta|-x_0\rangle \quad (1.28)$$

$$\rho_{partcula} = \begin{pmatrix} |\alpha|^2 & \alpha\beta^* \\ \beta\alpha^* & |\beta|^2 \end{pmatrix} \quad (1.29)$$

El sistema será también una superposición de estados:

$$|\Psi_{sistema}\rangle = \sqrt{P}(\alpha|x_0\rangle|1\rangle + \beta|-x_0\rangle|2\rangle) + \sqrt{1-P}(\beta|-x_0\rangle + \alpha|x_0\rangle)|0\rangle \quad (1.30)$$

con su correspondiente matriz 4x4:

$$\rho_{sistema} = |\Psi_{sistema}\rangle\langle\Psi_{sistema}| \quad (1.31)$$

Al realizar la traza parcial sobre el subespacio del fotón se obtiene la siguiente matriz densidad de la partícula:

$$\rho_{partcula} = \begin{pmatrix} |\alpha|^2 & \alpha\beta^*(1-P) \\ \beta\alpha^*(1-P) & |\beta|^2 \end{pmatrix} \quad (1.32)$$

Como se puede apreciar, los elementos de la diagonal no han variado al no variar la energía de la partícula como hemos supuesto al comienzo, no obstante los términos de fuera de la diagonal se han hecho más pequeños. Iterando, para un scattering con  $n$  fotones se obtiene:

$$\rho_n = \begin{pmatrix} |\alpha|^2 & \alpha\beta^*(1-P)^n \\ \beta\alpha^*(1-P)^n & |\beta|^2 \end{pmatrix} \quad (1.33)$$

Es de interés estudiar el término que aparece debido a la interacción con los fotones,  $(1-P)^n$ .  $P$  es la probabilidad de interacción, podemos expresar  $P$  en función de los pasos de tiempo

mediante una constante  $\Gamma$ , por otro lado  $n$  es el número de interacciones y también puede ser discretizado en función del tiempo  $n = t/\Delta t$  pudiendo reescribir el factor de interacción de la siguiente manera :

$$(1 - P)^n = (1 - \Gamma\Delta t)^{t/\Delta t} \quad (1.34)$$

Haciendo infinitesimal el tiempo entre interacciones  $\Delta t \rightarrow 0$  obtenemos:

$$(1 - P)^n = e^{-\Gamma t} \quad (1.35)$$

Donde la exponencial se obtiene resolviendo el límite con la regla de l'hôpital:

$$\lim_{\Delta t \rightarrow 0} (1 - \Gamma t)^{t/\Delta t} = 1^\infty \quad (1.36)$$

$$\lim_{\Delta t \rightarrow 0} \ln(1 - \Gamma t)^{t/\Delta t} = \lim_{\Delta t \rightarrow 0} \frac{-t\Gamma}{1 - \Gamma\Delta t} = \lim_{\Delta t \rightarrow 0} \frac{-\Gamma t}{1 - \Gamma\Delta t} = -\Gamma t \quad (1.37)$$

y por tanto:

$$\lim_{\Delta t \rightarrow 0} (1 - \Gamma\Delta t)^{t/\Delta t} = e^{-\Gamma t} \quad (1.38)$$

Con este resultado podemos expresar la matriz densidad (1.33) en función del tiempo:

$$\rho_{partcula} = \begin{pmatrix} |\alpha|^2 & \alpha\beta^*e^{-\Gamma t} \\ \beta\alpha^*e^{-\Gamma t} & |\beta|^2 \end{pmatrix} \quad (1.39)$$

De esta expresión se pueden sacar dos conclusiones fundamentales:

- Los elementos de fuera de la diagonal tienden a 0 exponencialmente con el tiempo
- Los elementos de la diagonal no varían

El primer punto es realmente el objetivo de este desarrollo, el hecho de que los elementos de fuera de la diagonal se anulen exponencialmente con el tiempo significa, según lo que he explicado con anterioridad, que el estado de la partícula se hace mixto con el tiempo, y por lo tanto el sistema se vuelve clásico, perdiendo las propiedades de superposición inicial.

Queda entonces ilustrado el gran problema que afronta la construcción de un ordenador cuántico. Deseamos crear maquinas que generen superposición y entrelazamiento, pero estos sistemas no pueden existir completamente aislados del entorno, y la interacción con el mismo les hace perder las propiedades por las que fueron construidos.

Sobre el segundo punto, simplemente quiero comentar que es consecuencia directa de que el modelo aquí presentado no considera que la partícula varía su energía o estado durante el proceso de interacción con el fotón. Existe un modelo más elaborado que no presentaré aquí, por motivos de espacio y por no alargar excesivamente esta ilustración, que considera que la interacción con el fotón puede dar lugar a procesos de emisión o absorción, excitando o desexcitando a la partícula. En este modelo, llamado Amplitude Damping Channel, la energía de la partícula varía, variando por tanto su estado y en consecuencia se puede observar como los elementos diagonales de la matriz densidad de la partícula sí varían. La variación de los estados de la partícula medida como consecuencia de las interacciones supone otra fuente de ruido.

## Capítulo 2

# Computación cuántica

### 2.1. Introducción a la computación cuántica

#### 2.1.1. ¿Por qué un ordenador cuántico?

Un ordenador cuántico es una máquina que funciona según las leyes de la Mecánica Cuántica y que por tanto, es capaz de aprovechar propiedades intrínsecamente cuánticas como el entrelazamiento y la superposición.

Un ordenador cuyo funcionamiento se basara en las propiedades anteriores, presentaría un alto grado de paralelismo, esto es, sería capaz de llevar a cabo múltiples tareas de forma simultánea, a diferencia de los ordenadores clásicos, cuyo funcionamiento es secuencial. Esto podría suponer una reducción abismal en el tiempo de cálculo y resolución de problemas, como se ha confirmado en los algoritmos de Grover y Shor. En 2001 se dió un salto importante en el campo de la computación cuántica cuando el algoritmo de Shor fue implementado y comprobado en la primera máquina cuántica, un ordenador de 7 qubits de IBM. En la figura 2.1, puede verse una comparación entre el algoritmo de Shor y un método clásico extraída de [1].

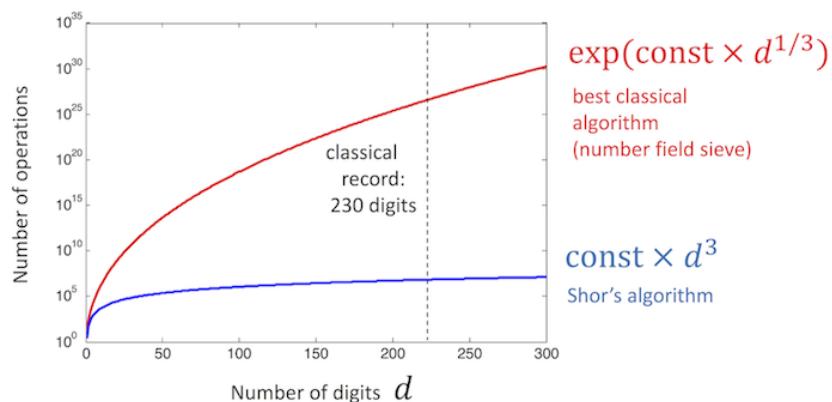


Figura 2.1: Reducción a tiempo polinómico del problema de factorización, mediante el algoritmo de Shor, y comparación con algoritmo clásico más optimizado.

Sistemas de encriptación como RSA, basan su seguridad en la necesidad de tiempo de computo del orden de millones de años para su resolución. El algoritmo de Shor, hasta entonces, un método puramente teórico, permitía convertir este tiempo exponencial en polinómico, haciendo posible la desencriptación. Este es un ejemplo realmente visual de hasta donde puede llegar la propiedad de paralelismo y en general, las capacidades de esta nueva tecnología.

### 2.1.2. Concepto de qubit

El qubit es la cuantización del bit. Un ordenador clásico, trabaja con una unidad básica, el bit, que puede tomar los valores 1, 0. Un ordenador cuántico sin embargo, utiliza como unidad fundamental el qubit, un objeto matemático que puede encontrarse en dos posibles estados,  $|0\rangle$  y  $|1\rangle$  y también en cualquier combinación lineal de los mismos, es decir, puede existir en un continuo de estados entre  $|0\rangle$  y  $|1\rangle$ :

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle \quad (2.1)$$

La condición  $|\alpha|^2 + |\beta|^2 = 1$  nos permite reescribir la expresión anterior de la forma siguiente:

$$|\Psi\rangle = \cos \frac{\theta}{2}|0\rangle + e^{i\varphi} \sin \frac{\theta}{2}|1\rangle \quad (2.2)$$

$\varphi$  y  $\theta$  describen un punto en la esfera unitaria que se puede ver en la imagen 2.2. Esta esfera recibe el nombre de esfera de Bloch y nos proporciona una forma visual de entender los estados de un qubit. También permite describir en ella las operaciones realizadas mediante las puertas de las que hablaré más adelante.

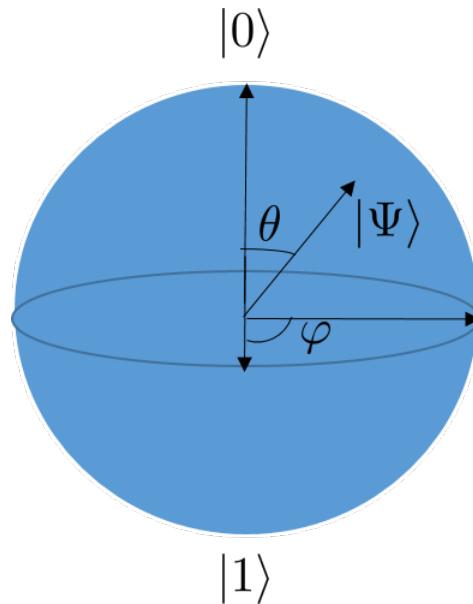


Figura 2.2: Representación de un qubit en la Esfera de Bloch

### 2.1.3. Puertas Cuánticas

Siguiendo con la analogía clásica, un ordenador realiza operaciones mediante el uso de puertas lógicas mientras que un ordenador cuántico realiza operaciones a través de puertas cuánticas. Estas puertas cuánticas deben poder implementarse y por tanto, no puede contradecir las leyes de la física. La evolución temporal de un estado viene descrita en mecánica cuántica por el postulado de evolución, que dicta que para cualquier evolución de un sistema, existe un operador  $U$  que transforma el sistema de la forma:

$$|\Psi_{final}\rangle = U|\Psi_{inicial}\rangle \quad (2.3)$$

El operador evolución es un operador unitario, por tanto, las puertas deben ser unitarias. Otra ley física a tener en cuenta, es la restricción dada por el teorema de no clonado, una restricción que impide obtener y utilizar copias de un estado y nos recuerda que trabajamos en un problema físico y no en buscar una mera transformación matemática que nos lleve de un estado a otro. Las puertas pueden actuar simultáneamente en uno o más qubits, siendo en realidad, operadores en un espacio de Hilbert teniendo una representación matricial. La imagen 2.3 muestra un ejemplo de los diagramas habituales de la computación cuántica donde aparecen los distintos qubits en su estado inicial. Las líneas simbolizan el paso del tiempo y sobre ellas están las puertas, que rigen la evolución temporal de los qubits hasta que finalmente son medidos.

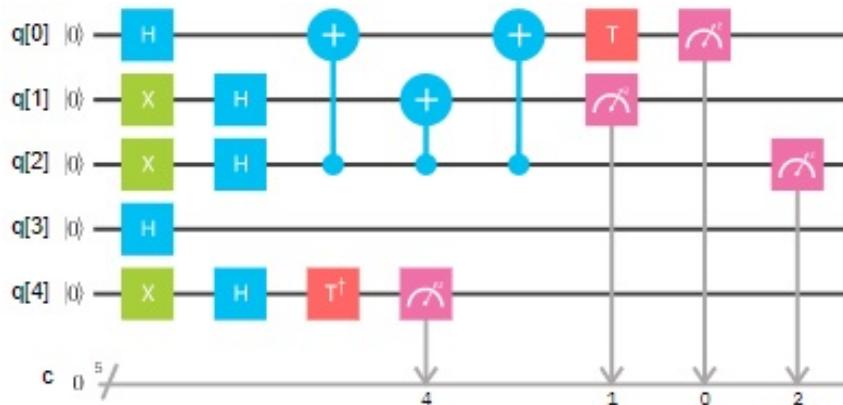


Figura 2.3: Diagrama en un ordenador de 5 qubits (IBM)

Una propiedad importante a mencionar es que el conjunto de puertas de 1 qubit y la puerta CNOT de dos qubits, forman un conjunto universal, esto es, que cualquier operación unitaria posible puede ser expresada aproximadamente con una secuencia finita de puertas del conjunto.

A continuación, en las imágenes 2.4 2.5 hago un breve compendio de las puertas que han sido utilizadas en este trabajo. Nótese que todas ellas son unitarias cumpliéndose que  $MM^\dagger = M^\dagger M = I$ .

### Puertas de 1 qubit

Hadamard		Inicial	$ 0\rangle$	$\frac{1}{\sqrt{2}}( 0\rangle +  1\rangle)$	Final	$\frac{1}{\sqrt{2}}( 0\rangle -  1\rangle)$	$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$
Puerta X		Inicial	$ 0\rangle$	$ 1\rangle$	Final	$ 0\rangle$	$ 1\rangle$
Puerta Y		Inicial	$ 0\rangle$	$i 1\rangle$	Final	$-i 0\rangle$	$ 1\rangle$
Puerta Z		Inicial	$ 0\rangle$	$ 1\rangle$	Final	$ 0\rangle$	$- 1\rangle$
Puerta S		Inicial	$ 0\rangle$	$ 1\rangle$	Final	$ 0\rangle$	$i 0\rangle$
Puerta T		Inicial	$ 0\rangle$	$ 1\rangle$	Final	$ 0\rangle$	$e^{i\pi/4} 1\rangle$

Figura 2.4: Símbolo, actuación y representación matricial de las puertas de 1 qubit

### Una importante puerta de 2 qubits: Puerta CNOT

La puerta CNOT posee un primer qubit de control que regula la aplicación de una puerta X en el segundo qubit.

CNOT		inicial		final	
		0\rangle	0\rangle	0\rangle	0\rangle
		0\rangle	1\rangle	0\rangle	1\rangle
		1\rangle	0\rangle	1\rangle	1\rangle
		1\rangle	1\rangle	1\rangle	0\rangle

$$I_2 \otimes \sigma_x = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Figura 2.5: Símbolo, actuación y representación matricial de la puerta CNOT

## 2.2. Implementación física de puertas: Oscilaciones de Rabi

### 2.2.1. Puertas de 1 qubit

En este apartado busco explicar como se construyen los componentes fundamentales que describen las operaciones realizadas sobre los qubits, las puertas. Existen diferentes formas de implementar físicamente un qubit, una de ellas consiste en usar espines. Los espines tienen transiciones energéticas que pueden ser estimuladas con un campo, de forma que esta actuación de los campos daría lugar a las puertas. Es en este proceso de estimulación de transiciones donde entran en juego las oscilaciones de Rabi, generando las puertas.

Un Hamiltoniano independiente del tiempo muy habitual es el siguiente:

$$H_0 = \frac{\Delta\sigma_z}{2} \quad (2.4)$$

Se trata de un Hamiltoniano constante, que representa un sistema de dos niveles de diferencia de energías  $\Delta$ .

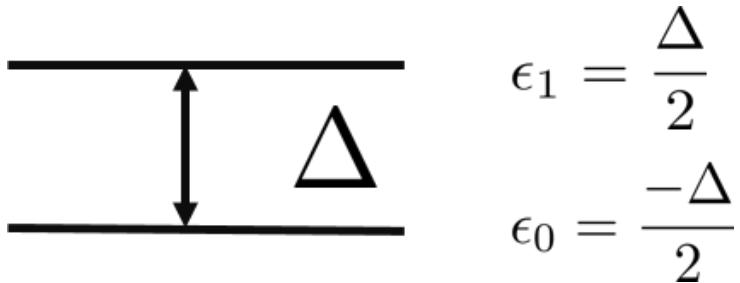


Figura 2.6: Niveles de energía del Hamiltoniano

A este Hamiltoniano se le puede añadir un término dependiente del tiempo :

$$H = H_0 + H_1 = \frac{\Delta\sigma_z}{2} + 2\lambda \cos(\omega t)\sigma_x \quad (2.5)$$

Este término es lo que se conoce como un *drive*, una estimulación con un campo variable con el tiempo, que produce las llamadas oscilaciones de Rabi. Controlando estas oscilaciones podemos generar superposiciones de estados  $|0\rangle$  y  $|1\rangle$ , y por tanto implementar puertas de 1 qubit. Como se muestra en el apéndice, la evolución temporal dada por la ecuación de Schrödinger en el caso particular de las condiciones iniciales indicadas en (2.7), para una función de ondas de la forma de (2.6) permite llegar a las ecuaciones (2.8)(2.9), que no son soluciones exactas, sino que se obtienen al aplicar la aproximación RWA que desprecia los términos exponenciales que oscilan muy rápidamente.

$$|\Psi\rangle = c_1(t)|0\rangle + c_2(t)|1\rangle \quad (2.6)$$

$$c_1(0) = 1 \quad c_2(0) = 0 \quad (2.7)$$

$$|c_1(t)|^2 = \frac{\delta^2}{\Omega^2} + \left(\frac{2\lambda}{\Omega}\right)^2 \cos^2 \frac{\Omega t}{2} \quad (2.8)$$

$$|c_2(t)|^2 = \left(\frac{2\lambda}{\Omega}\right)^2 \sin^2 \frac{\Omega t}{2} \quad (2.9)$$

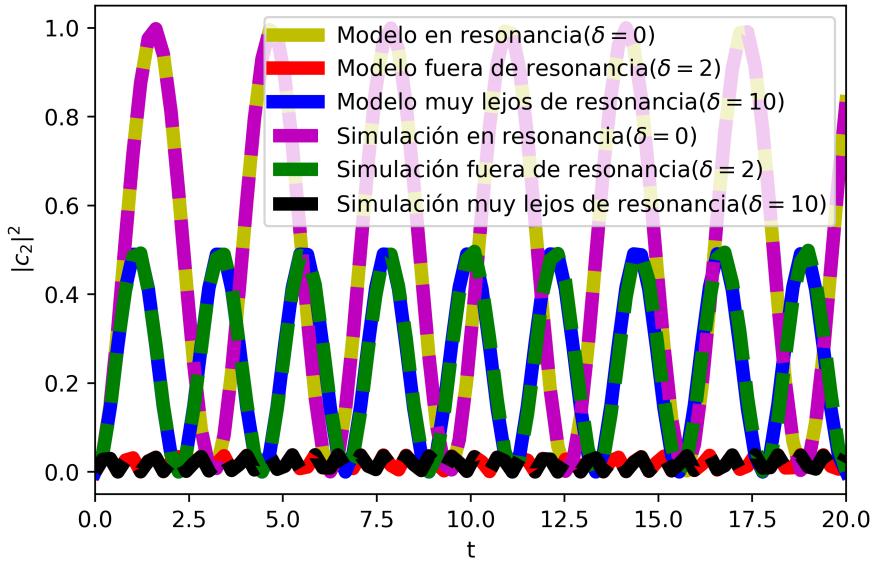
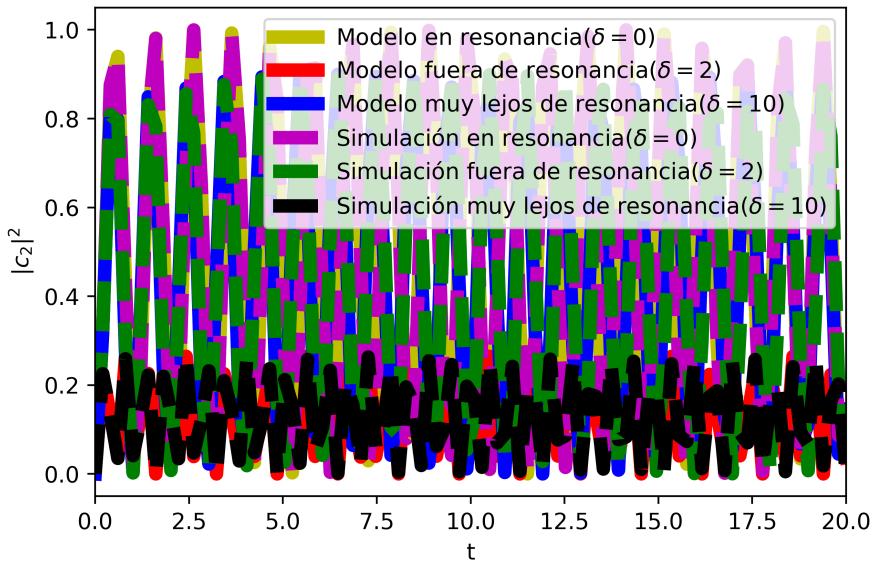
Que en resonancia,  $\delta = 0$ , se simplifican llegando a (2.10)(2.11):

$$|c_1(t)|^2 = \cos^2 \frac{\Omega t}{2} \quad (2.10)$$

$$|c_2(t)|^2 = \sin^2 \frac{\Omega t}{2} \quad (2.11)$$

Para los valores  $t = 0, \frac{2\pi}{\Omega}$  la probabilidad de estar en el estado  $|0\rangle$  es 1, estos valores determinan un ciclo, por otro lado la probabilidad del estado  $|1\rangle$  es 1 para  $t = \frac{\pi}{\Omega}$ . Es decir, driveando correctamente, se puede generar superposición entre el estado  $|0\rangle$  y el estado  $|1\rangle$  y por lo tanto implementar puertas. Por ejemplo, un drive a lo largo de un intervalo completo mantiene el qubit invariante, si se toman valores dentro de este intervalo podemos generar superposición, en concreto para  $t = \frac{\pi}{2\Omega}$  los coeficientes son iguales y se tiene una puerta Hadamard. Un pulso de  $t = \frac{\pi}{\Omega}$  invierte el qubit (esto es llamado pulso  $\pi$ ), lo que permite implementar un puerta X. Las oscilaciones de Rabi solo producen cambios de coeficientes reales, para generar el resto de puertas de la figura 2.4, que introducen una fase imaginaria es necesario drivear con términos que incorporen  $\sigma_y$  en el coseno de la expresión (2.6), así se pueden generar puertas como la puerta Y, o la puerta de fase. Genericamente este término puede escribirse como un sumatorio sobre  $\sigma_x, \sigma_y$  y  $\sigma_z$  ya que las tres sigmas de Pauli son los generadores de rotaciones en SU(2) y pueden generar cualquier rotación en la esfera de Bloch.

He realizado la evolución de un estado  $|0\rangle$  mediante el Hamiltoniano indicado en (2.5), utilizando python y la herramienta qutip, este resultado ha sido comparado con la ecuación (2.9) y efectivamente se comprueba que la evolución de un qubit sigue el comportamiento de las ecuaciones obtenidas y por tanto estas permiten explicar la evolución temporal de un qubit. Los resultados se presentan en la gráfica 2.7, donde se observan las oscilaciones de Rabi completas para frecuencia resonante, y como conforme se alejan de la resonancia van reduciéndose hasta anularse y producir un cambio nulo en el qubit. Los datos vienen expresados en unidades de  $f = \frac{2\pi}{\omega} = 1$ . Por otro lado, es interesante comprobar la validez de las soluciones aproximadas (2.8)(2.9) en la figura 2.8 puede observarse como si se aumenta la amplitud y se ignora la condición de la aproximación RWA, las ecuaciones (que han sido obtenidas en el apéndice utilizando esta aproximación para poder resolver las ecuaciones diferenciales que nos deja la ecuación de Schrödinger) dejan de funcionar. También se puede observar como esto es mucho más notorio fuera de resonancia, ya que en resonancia una de las dos exponenciales que no se desprecian, tiene contribución máxima y no oscila, ( $e^{-i(\omega-\omega_0)t} = e^{-i(0)t} = 1$ ), de forma que el término que oscila rápidamente ( $e^{i(\omega+\omega_0)t}$ ) es aún más despreciable..

Figura 2.7: Oscilaciones de Rabi dentro de la aproximación RWA ( $\lambda = 1, \omega = 300$ )Figura 2.8: Oscilaciones de Rabi fuera de RWA( $\lambda = 3, \omega = 300$ )

### 2.2.2. Puertas de 2 qubits

La implementación de puertas de orden superior se basa de igual forma, en las oscilaciones de Rabi para producir cambios en los qubits. En concreto, en este apartado explicaré la imple-

mentación de la puerta CNOT, no obstante, toda puerta de 2 qubits se puede implementar de la misma forma. Una puerta de 2 qubits, como es evidente, requiere a diferencia del caso anterior dos qubits y además requerirá de la existencia de una interacción entre ambos. Es por esto que el Hamiltoniano que describa dicho sistema, tendrá la siguiente forma:

$$H_{2qubit} = \sigma_z^{(1)} + \epsilon \sigma_z^{(2)} + J \sigma_z^{(1)} \sigma_z^{(2)} \quad (2.12)$$

Donde los dos primeros términos hacen referencia a los qubit, y el tercero a la interacción entre estos. Si se calculan los autovalores del Hamiltoniano, se puede realizar el diagrama de niveles energéticos de la figura 2.9 y encontrar las distintas transiciones entre niveles.

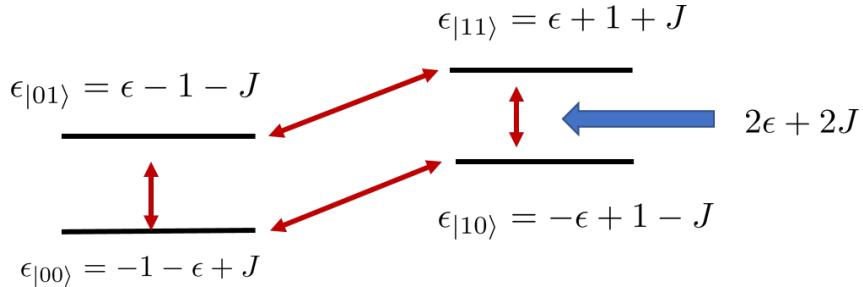


Figura 2.9: Niveles de energía y transiciones del Hamiltoniano  $H_{2qubit}$

Ahora se trata de estimular las transiciones deseadas mediante un driving. En este caso, la transición que permite implementar la puerta CNOT es la señalada en la imagen 2.9 (la tabla de actuación de la puerta CNOT estaba en la imagen 2.5). Para ello se realiza un drive con una frecuencia resonante, análogamente a como se mostró en las sección 2.2.1, para estimular únicamente esta transición:

$$\Delta = 2\epsilon + 2J \quad (2.13)$$

La matriz que representa este cambio de estado  $|10\rangle$  a  $|11\rangle$  y viceversa es  $\sigma_x$ . Así que el drive serán oscilaciones de Rabi de la forma:

$$H_{drive} \sim \sigma_x \cos \Delta t \quad (2.14)$$

Con esto se ha obtenido una explicación de como obtener una puerta de 2 qubits, en concreto una CNOT. Esto permite concluir diciendo que las puertas de 1 qubit, suponen un sistema de dos niveles donde se realiza un drive entre los mismos, mientras que, para la puerta CNOT, se tiene un modelo de Ising driveado.

# Capítulo 3

## IBM Quantum experience

### 3.1. ¿Qué es IBM Quantum Experience?

Quantum experience es un proyecto sacado al público por IBM. Gracias a este proyecto investigadores de todo el mundo pueden hacer uso de los prototipos de ordenadores cuánticos desarrollados por la empresa. Esto supone que los usuarios podemos formar parte de esta gran revolución científica y tecnológica mientras que, IBM puede poner a prueba sus ordenadores en un proceso de mejora y desarrollo. Quantum experience no es solo esto, es también una carrera por desarrollar e implementar las ideas responsables de una posible nueva era científica y tecnológica.

### 3.2. Dentro de la Quantum Experience

Para poder acceder a este proyecto es necesario crear una cuenta en la quantum experience. Existen distintas herramientas para poder acceder a los ordenadores, pero la principal es el “composer”, una interfaz gráfica que muestra las distintas máquinas con las que se pueden trabajar y datos realmente interesantes como la estructura de los qubits, el error asociado a cada qubit y la temperatura de refrigeración.

Desde el composer se puede acceder al simulador y a los ordenadores. Aparte del composer también existe un foro que resulta crucial dentro del proyecto al estar este basado en la cooperación de investigadores de todo el mundo a través de la nube.

Junto al foro también es destacable una página de noticias que informa de los distintos avances y que además recopila una gran cantidad de fotos donde se enseña la estructura física de los ordenadores, manuales para los investigadores que no conocemos temas avanzados o métodos para implementar experimentos, y para los que se inician en el terreno de la mecánica cuántica, así como vídeos grabados por el equipo de IBM explicando el funcionamiento interno y más detalles sobre los experimentos.

Toda cuenta tiene dentro del composer acceso al simulador, un servidor donde el código es ejecutado en un ordenador clásico para obtener unos datos teóricos correctos que sirvan para comparar posteriormente.

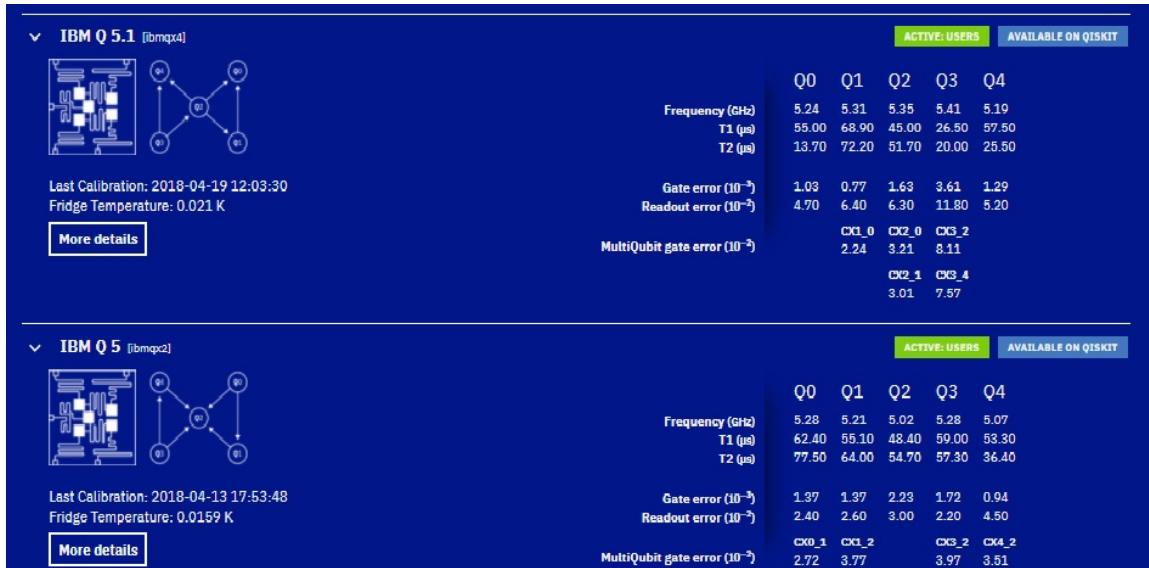


Figura 3.1: Ordenadores de 5 qubits y datos de interés en la interfaz del composer

Existen varios modelos de funcionamiento de un ordenador cuántico, sin embargo, los prototipos por los que se está apostando actualmente son principalmente dos, ordenadores de trampa de iones y de superconductores. Los ordenadores de la Quantum Experience están basados en una tecnología de superconductores y tienen una topología llamada *star-shaped*, esto significa que su conectividad está limitada a primeros vecinos. En la figura 3.2 he representado este tipo de conectividad, en contraste, con una topología de conectividad total.

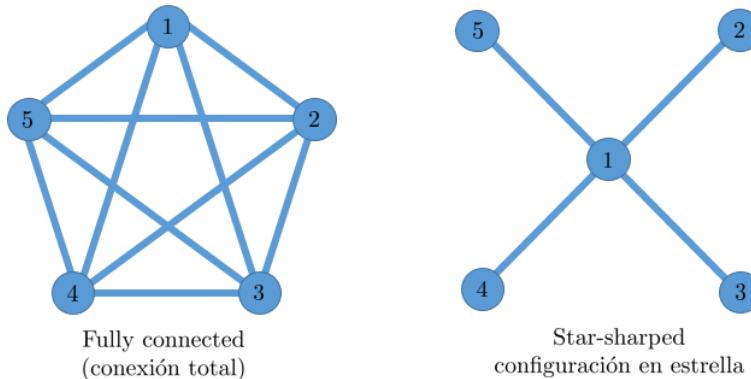


Figura 3.2: A la derecha, la topología en estrella de los ordenadores de IBM que he utilizado, 4 puertas CNOT conectan 4 de los qubits a un qubit central. A la izquierda una topología de conectividad total

### 3.3. Un sistema para usar mecánica cuántica en un ordenador

¿Cómo hacer que un ordenador entienda un algoritmo cuántico?, la respuesta es que se requiere incorporar una serie de herramientas para poder crear y compilar código y que este sea ejecutado en un ordenador cuántico real. Es por esto que se ha desarrollado el “Quantum Information Software kit” también llamado QISKit, el cual incorpora las herramientas necesarias para poder generar códigos usando la herramienta Jupyter y el lenguaje Python. Una de las grandes novedades de este kit es la incorporación de un sub-lenguaje de programación completamente nuevo, que permite explicar usando python, las especificaciones que se requieren para trabajar con un ordenador cuántico, como por ejemplo asignar las operaciones a los distintos qubits.

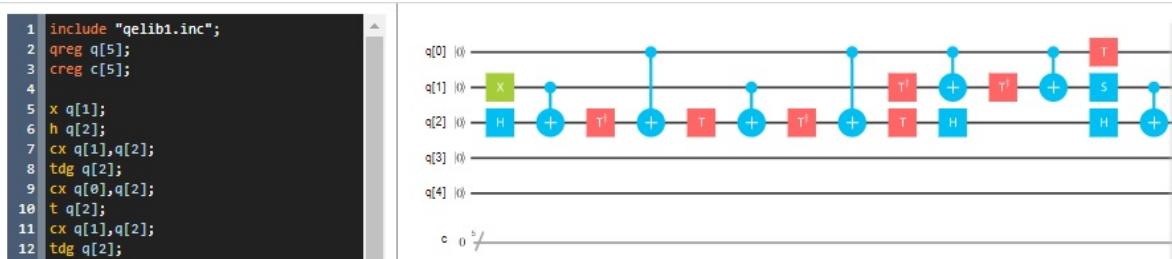


Figura 3.3: Simulación de un circuito cuántico en la interfaz gráfica del composer y código del circuito.

Mediante este sub-lenguaje y el uso de QISKit se puede hacer correr algoritmos en el simulador y en los ordenadores cuánticos de IBM, que es al fin y al cabo, el núcleo central de todo este proyecto. Para esto, toda cuenta tiene asignadas unas unidades llamadas tokens que funcionan como una moneda para poder realizar ejecuciones en los ordenadores reales. Cada ejecución cuesta una cantidad concreta de tokens que se recuperan cuando la ejecución ha finalizado y se entregan los resultados, de forma que esto simplemente limita la cantidad de ejecuciones simultáneas que se realizan. Los ordenadores que he utilizado son ordenadores de 5 qubits, aunque durante este trimestre IBM ha abierto al público la posibilidad de utilizar un ordenador de 16 qubits.



## Capítulo 4

# Resultados obtenidos de un ordenador cuántico

### 4.1. Estudio de la decoherencia

Uno de los objetivos más interesantes que ha tenido el uso de un ordenador cuántico real en este trabajo ha sido la comprobación del fenómeno de decoherencia. Recordando la sección 1.3, el fenómeno de decoherencia se manifiesta como una reducción exponencial de los términos de fuera de la diagonal de la matriz densidad del sistema.

La matriz de un estado superposición, como el obtenido por aplicación de una puerta Hadamard, tiene la siguiente forma:

$$\rho = \begin{pmatrix} |\alpha|^2 & \gamma^* \\ \gamma & |\beta|^2 \end{pmatrix} \quad (4.1)$$

Al aplicar una puerta Hadamard sobre el estado  $|0\rangle$  para realizar superposición, se obtiene que teóricamente la matriz viene dada por los siguientes valores:

$$\rho = \begin{pmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{pmatrix} \quad (4.2)$$

El proceso de decoherencia consistirá entonces, en que los elementos no diagonales obtenidos serán inferiores al calculo teórico. No obstante, las medidas que se obtienen al ejecutar un algoritmo son únicamente los valores diagonales. Es por esto que es necesario crear un algoritmo que permita obtener los valores no diagonales mediante un sistema de ecuaciones. Para esto he ejecutado dos algoritmos sencillos que permiten obtener un sistema de 2 ecuaciones, para calcular los valores deseados, a partir de las medidas tomadas del ordenador. Estos algoritmos se pueden ver en la figura 4.1 y consisten en:

- Un Circuito de un qubit con dos puertas Hadamard
- Un Circuito de un qubit con dos puertas Hadamard y con la aplicación de una puerta S entre ambas.

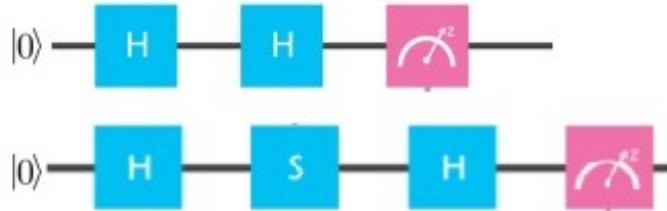


Figura 4.1: Algoritmos diseñados para obtener las partes real e imaginaria de la decoherencia mediante un sistema de dos ecuaciones

Se puede comprobar que la aplicación de estas puertas (cuya representación se encontraba en la sección 2.1.3, figura 2.2.1) sobre la matriz del sistema inicial (4.1), da lugar a matrices, cuyos elementos diagonales son suma de todos los elementos de la matriz inicial, de forma que, la diagonal de la matriz obtenida al aplicar las puertas de uno de los algoritmos, es una ecuación en función de las incógnitas  $\gamma$  y  $\gamma^*$ . Para obtener una ecuación más, hago uso del segundo algoritmo, de forma que se tiene un sistema de 2 ecuaciones con dos incógnitas. Esto me permite obtener la parte real y la parte imaginaria de  $\gamma$  contando así, con los elementos matriciales que no conocía de la matriz densidad (4.1). Los resultados vienen expresados en la figura 4.2.

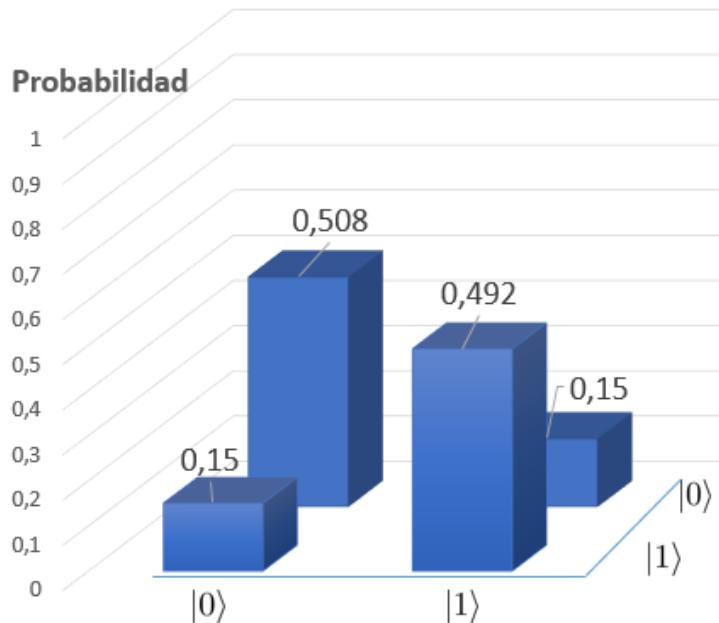


Figura 4.2: Elementos de la matriz densidad del estado superpuesto, medidos en un ordenador cuántico real

Como se puede observar, los elementos de la matriz densidad del sistema no se corresponden con los calculados teóricamente en (4.1). De hecho, se observa que mientras los valores diagonales son los esperados, los valores de fuera de la diagonal son muy inferiores a los calculados. Esto es debido a que a lo largo del proceso de ejecución la interacción con el medio los hace tender a

cero exponencialmente con el tiempo. De esta forma, el diseño de dos algoritmos cuánticos y la toma de medidas en el ordenador ha permitido observar la perdida de fidelidad por decoherencia, ilustrando este proceso.

## 4.2. Implementación y simulación de un puerta Toffoli

En esta sección quiero mostrar los resultados obtenidos de la aplicación de una puerta Toffoli, esto además me va a permitir ilustrar con datos reales otra de las principales fuentes de error, el uso de un número de puertas elevado. La puerta Toffoli es una puerta de 3 qubits que aplica en los dos primeros una operación de control y en el tercero una puerta X. En la figura 4.3 pueden verse sus principales características, entre ellas los dos únicos estados que no permanecen invariantes bajo su actuación.



Figura 4.3: Símbolo, actuación y representación matricial de la puerta Toffoli

En la quantum experience, cualquier circuito ha de ser implementado a partir de puertas de 1 y 2 qubits, de forma que para poder estudiar esta puerta he tenido que implementarla con el algoritmo de la figura 4.4, el cual puede encontrarse en [2]. La imagen corresponde a la implementación del algoritmo en el ordenador cuántico, para ello he comenzado con dos puertas X para generar un estado inicial  $|110\rangle$ .

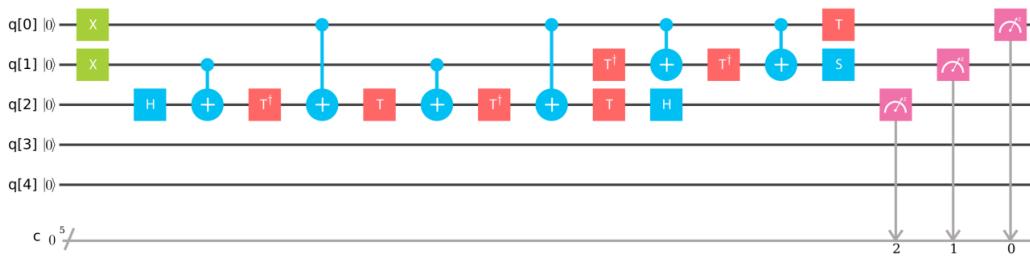


Figura 4.4: Construcción de la puerta Toffoli a partir de puertas de 1 y 2 qubit

El funcionamiento de este sistema de puertas es complejo, en primer lugar las puertas Hadamard crean una superposición de estados. Sobre esta superposición actúan las puertas  $T$  y  $T^\dagger$  que añaden desfases a los estados, siendo las puertas CNOT las encargadas de seleccionar que estados

reciben desfase  $\pi/4, -\pi/4$  o no reciben desfase, de forma que al final los estados pueden quedar con desfase  $\pm\pi/2$  o con desfase nulo. Los estados que no reciben fase permanecen invariantes, los demás pierden su carácter imaginario con la aplicación de la puerta  $S$ . (Los detalles de las puertas se encontraban en la sección 2.1.3 en la figura 2.4)

Es decir el mecanismo funciona debido a la generación de estados superpuestos, cuyas componentes se van seleccionando mediante las puertas CNOT y los desfases hasta dejar finalmente invariantes a 6 estados y cambiando a dos de ellos.

Resulta entonces interesante tomar un estado  $|110\rangle$  y comprobar que se transforma en un estado  $|111\rangle$  bajo actuación de la puerta Toffoli. También resulta de interés, aplicar dos puertas Toffoli secuencialmente para comprobar que se recupera el estado inicial. Dado que el proceso de decoherencia introduce un error aleatorio, he ejecutado 10 veces cada algoritmo, para hacer estadística. Los resultados obtenidos pueden verse en los histogramas 4.5 4.6

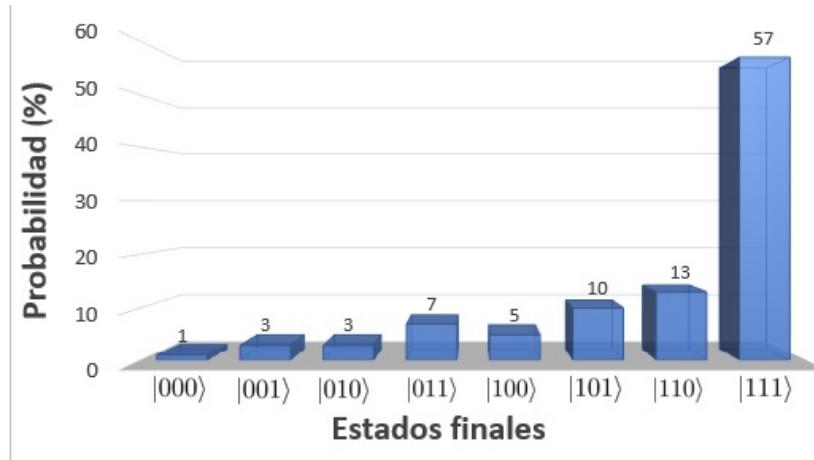


Figura 4.5: Distribución de estados obtenida de la aplicación de una puerta Toffoli

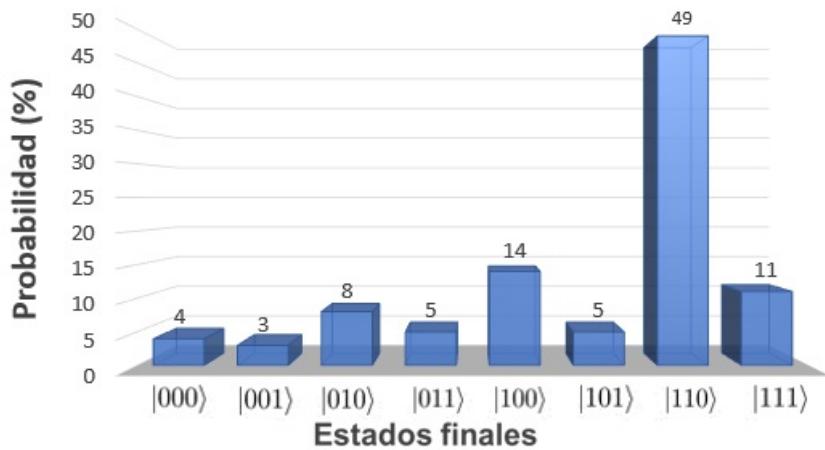


Figura 4.6: Distribución de estados obtenida de la aplicación de dos puertas Toffoli

Como puede verse, los resultados se alejan de lo esperado en gran medida, las transformaciones solo tienen lugar correctamente el 57% y el 49% de las veces respectivamente. En su lugar, se llevan a cabo otras transformaciones que no deberían darse, de forma que el resultado final convierte nuestro estado inicial, en una mezcla de todos los estados posibles. Esto es comprensible al tratarse de unos prototipos de una tecnología en desarrollo, pero el principal motivo es el uso de 18 puertas. Como he comentado anteriormente, el uso de muchas puertas es una fuente de error importante, existen ordenadores que alcanzan una gran fidelidad al implementar una puerta, no obstante, cuando se implementan algoritmos de unas pocas puertas el error es inmenso. Esto me permite concluir con que el número de puertas utilizado es realmente significativo en cuanto al error generado.

### 4.3. Conclusiones

- Como se muestra en la metodología aplicada en el estudio de la decoherencia (sección 4.1), la construcción de ordenadores cuánticos, y la apertura de su uso al público mediante IBM Quantum Experience, es un gran avance para la computación cuántica, tratándose de una herramienta que permite avanzar en el desarrollo de esta tecnología y facilitar el estudio de los procesos de error que son el principal problema de su implementación.
- Las fidelidades que proporcionan los ordenadores de IBM Quantum Experience, son todavía muy limitadas, debido a que son prototipos de una tecnología que todavía está en desarrollo.
- El número de qubits y la cantidad de puertas que se pueden implementar manteniendo el error en un rango razonable es reducida.
- El proceso de decoherencia supone una fuente de error considerable, y el avance de esta tecnología depende de desarrollar mecanismos de corrección del mismo.



# Bibliografía

- [1] *IBM, Shor algorithm.* [https://quantumexperience.ng.bluemix.net/proxy/tutorial/full-user-guide/004-Quantum\\_Algorithms/110-Shor's\\_algorithm.html](https://quantumexperience.ng.bluemix.net/proxy/tutorial/full-user-guide/004-Quantum_Algorithms/110-Shor's_algorithm.html),
- [2] Michael A Nielsen e Isaac L Chuang. *Quantum computation and quantum information.* 2000.
- [3] Thomas Dittrich y col. *Quantum transport and dissipation.* Vol. 3. Wiley-Vch Weinheim, 1998.
- [4] Claude Cohen-Tannoudji, Bernard Diu y Franck Laloë. “Quantum mechanics, vol. I”. En: *I (Hermann, Paris, 1977)* (1977).
- [5] Diego García-Martín y Germán Sierra. “Five Experimental Tests on the 5-Qubit IBM Quantum Computer”. En: *arXiv preprint arXiv:1712.05642* (2017).
- [6] Norbert M Linke y col. “Experimental comparison of two quantum computing architectures”. En: *Proceedings of the National Academy of Sciences* (2017), pág. 201618020.
- [7] *IBM Quantum Experience.* <https://quantumexperience.ng.bluemix.net/qx/community>,
- [8] Richard P Feynman. “Simulating physics with computers”. En: *International journal of theoretical physics* 21.6-7 (1982), págs. 467-488.



# Apéndice

## Oscilaciones de Rabi

Dado el Hamiltoniano (1)

$$H = H_0 + H_1 = \frac{\Delta\sigma_z}{2} + 2\lambda \cos(\omega t)\sigma_x \quad (1)$$

La evolución de una función de onda de la forma (2), mediante la ecuación de Schrödinger, da lugar, sacando la fase  $e^{i\Delta t/2\hbar}$  a la ecuación (3)

$$\Psi = \begin{pmatrix} c_1(t) \\ c_2(t) \end{pmatrix} \quad (2)$$

$$i\hbar \frac{d}{dt} \begin{pmatrix} c_1(t)e^{i\Delta t/2\hbar} \\ c_2(t)e^{-i\Delta t/2\hbar} \end{pmatrix} = \begin{pmatrix} -\Delta/2 & -2\lambda\hbar \cos(\omega t) \\ -2\lambda\hbar \cos(\omega t) & \Delta/2 \end{pmatrix} \begin{pmatrix} c_1(t)e^{i\Delta t/2\hbar} \\ c_2(t)e^{-i\Delta t/2\hbar} \end{pmatrix} \quad (3)$$

Tomando  $\hbar\omega_0 = \Delta$  se obtienen el siguiente sistema de ecuaciones diferenciales:

$$\frac{dc_1}{dt} = i\lambda(e^{i(\omega-\omega_0)t} + e^{-i(\omega+\omega_0)t})c_2 \quad (4)$$

$$\frac{dc_2}{dt} = i\lambda(e^{-i(\omega-\omega_0)t} + e^{+i(\omega+\omega_0)t})c_1 \quad (5)$$

Se trata de un sistema de ecuaciones de segundo orden, lineales, homogéneas, de coeficientes variables. En ese punto, con el fin de simplificar las ecuaciones obtenidas hago uso de la aproximación RWA, de forma que asumiendo  $\lambda \ll \omega$  y  $\delta = \omega - \Delta = 0$  podemos despreciar los términos que oscilan muy rápido y reducir las ecuaciones a un sistema de coeficientes constantes (7):

$$\frac{dc_1}{dt} = i\lambda e^{i\delta t} c_2 \quad (6)$$

$$\frac{dc_2}{dt} = i\lambda e^{-i\delta t} c_1 \quad (7)$$

He obtenido las soluciones de este sistema para las condiciones de contorno  $c_1(0) = 1$ ,  $c_2(0) = 0$ , que corresponden a comenzar con un espín hacia arriba, dichas soluciones son las correspondientes a (9)(10), donde anteriormente he definido la frecuencia de Rabi como (8)

$$\Omega = (\delta^2 + 4\lambda^2)^{1/2} \quad (8)$$

$$c_1(t) = e^{i\delta t/2} \left( \cos \frac{\Omega t}{2} - i \frac{\delta}{\Omega} \sin \frac{\Omega t}{2} \right) \quad (9)$$

$$c_2(t) = e^{-i\delta t/2} \frac{2i\lambda}{\Omega} \sin \frac{\Omega t}{2} \quad (10)$$

Quiero hacer énfasis en que estas soluciones no son exactas sino que son soluciones aproximadas mediante la aproximación RWA. Más detalles sobre este desarrollo pueden ser consultados en [3]

## Código de python

```

#Inicializo variables
Psi0=basis(2,0)
tlist = linspace(0, 20, 100)
Lam=1                         #Amplitud Lambda
delta=10
Omega=sqrt (delta*delta+4*Lam*Lam) # Frecuencia de Rabi
om=300                         # Frecuencia
Delta=om-delta
Omegares=2*Lam                  #Frecuencia de Rabi en resonancia
Omegap=sqrt (2*2+4*Lam*Lam)     #Frecuencia de Rabi para delta=2
#Defino el Hamiltoniano (la dependencia con el tiempo tiene
una sintaxis especial)

H0=Delta*sigmaz()/2;
H1 =sigmax()
def H1_coeff(t ,arg ):
    return 2*Lam*cos (om*t)
h_t = [H0,[H1, H1_coeff ]]

# Defino una matriz tal que su valor esperado sea | c2 | ^2
modQuadC2=sigmax()*sigmap()

#Realizo la evolución temporal y calculo el valor esperado de modQuadC2 (delta
medata=mesolve(h_t, Psi0 , tlist , [] ,modQuadC2)

#Defino el mismo Hamiltoniano pero bajo condición de resonancia
H0=om*sigmaz()/2;
H1 =sigmax()
def H1_coeff(t ,arg ):
    return 2*Lam*cos (om*t)
h_res = [H0,[H1, H1_coeff ]]

```

```

#Calculo el valor de modQuadC2 en resonancia
meres=mesolve(h_res, Psi0, tlist, [], modQuadC2)

#Defino el mismo Hamiltoniano pero con delta=2
H0=98*sigmaz()/2;
H1=sigmax()
def H1_coeff(t, arg):
    return 2*Lam*cos(100*t)
h_res = [H0, [H1, H1_coeff]]

#Calculo el valor de modQuadC2 con delta=2
minimo=mesolve(h_res, Psi0, tlist, [], modQuadC2)

# Genero las im'agenes
SAVE=1;
Imagen=2;
fig, ax = plt.subplots()

# Represento las soluciones de Rabi para resonancia, delta=2 y delta=10
ax.plot(tlist,(2*Lam/Omegares)*(2*Lam/Omegares)*sin(Omegares*tlist/2)
         *sin(Omegares*tlist/2),'-y', label="Modelo en resonancia
         (" '$\delta=0$' , linewidth=6);
ax.plot(tlist,(2*Lam/Omega)*(2*Lam/Omega)*sin(Omega*tlist/2)
         *sin(Omega*tlist/2),'-r', label="Modelo fuera de resonancia
         (" '$\delta=2$' , linewidth=6);
ax.plot(tlist,(2*Lam/Omegap)*(2*Lam/Omegap)*sin(Omegap*tlist/2)
         *sin(Omegap*tlist/2),'-b', label="Modelo muy lejos de resonancia
         (" '$\delta=10$' , linewidth=6);

# Represento las evoluciones temporales para resonancia, delta=2 y delta=10
ax.plot(tlist,meres.expect[0],'-m', label="Simulaci'on en resonancia
         (" '$\delta=0$' , linewidth=6)
ax.plot(tlist,minimo.expect[0],'-g', label="Simulaci'on
         fuera de resonancia(" '$\delta=2$' , linewidth=6)
ax.plot(tlist,medata.expect[0],'-k', label="Simulaci'on
         muy lejos de resonancia(" '$\delta=10$' , linewidth=6)

ax.set_xlabel(r't');
ax.set_ylabel(r'$|c_2|^2$');
ax.set_xlim(0,20);
ax.legend(loc=1);
plt.show(fig)

```